

ESEIAAT

Trabajo de Final de Grado



UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

Escola Superior d'Enginyeries Industrial,
Aeroespacial i Audiovisual de Terrassa

Memoria:

**Proyecto y diseño de un sistema de guiado para un RPA de servicio
de transporte de mercancías puerta a puerta**

Estudiante: Marta Muñoz Delgado

Grado: Ingeniería en Vehículos Aeroespaciales

Fecha de entrega: 10 de junio de 2019

Dirigido por: Luis Manuel Pérez Llera

Agradecimientos

Me gustaría agradecer el apoyo de mi tutor Luis Manuel Pérez Llera. Él fue quien propuso este proyecto y quien me ha ayudado y guiado durante todos estos meses. Gracias por ayudarme a llevarlo a cabo.

También me gustaría dar las gracias a todas las personas que me han apoyado durante la realización de este trabajo, en especial, a los miembros del proyecto Aldora: Edu Gómez, Ferran Lumbierres, Edu Rius y Francesc Viaplana. Ellos me han ayudado desde el principio, me han dejado utilizar uno de sus drones para las pruebas de vuelo y, además, me han acompañado durante su realización.

Por último, quisiera agradecer a mi familia y amigos por estar a mi lado todos estos meses y por su apoyo incondicional.

Resumen

Este trabajo consiste en el proyecto y diseño de un sistema de guiado para un RPA de servicio de transporte de mercancías puerta a puerta. Para ello, se realizará una búsqueda de información sobre el estado actual de los dornes en el transporte de mercancías, se analizará la normativa vigente en el Estado Español para estudiar su posible implementación y se seleccionarán los sensores y actuadores que deberá incorporar el prototipo a construir.

Por lo que respecta a la programación, se escogerá un entorno de programación compatible con los componentes seleccionados y se realizará un programa para que el dron sea capaz de llevar a cabo un transporte desde un lugar de origen hasta unas coordenadas concretas. Posteriormente se validarán los programas en un simulador para asegurar su eficacia.

Por último, se construirá un prototipo con los componentes previamente seleccionados y se realizarán las pruebas de vuelo de los programas realizados.

Índice

Agradecimientos	2
Resumen	3
Declaración de honor	4
I Introducción	11
1. Objeto	11
2. Alcance	11
3. Requisitos básicos	11
4. Justificación	11
II Estado del arte	13
1. Amazon Prime Air	14
2. DHL Parcelcopter	15
3. GoogleX Project Wing	16
4. UPS	17
5. Boeing CAV	18
III Análisis de la normativa	19
1. Requisitos de la aeronave y los sistemas	19
2. Condiciones de vuelo	20
3. Pilotos y observadores	20
IV Sensores	21
1. Acelerómetro	21
2. Giroscopio	21
3. Barómetro	22
4. Magnetómetro	23
5. GPS	23
V Actuadores	25
1. Motor y hélice	25

2. Batería	26
3. Regulador de velocidad	26
4. Placa distribuidora	26
5. Controladora de vuelo	27
6. Ordenador de abordó	28
7. Sistema de radio control	28
8. Telemetría	28
 VI Elección de componentes	 30
1. Sistema motor-propulsor	30
1.1. Motor	30
1.2. Hélice	32
1.3. Batería	33
1.4. Regulador de velocidad	33
2. Sistemas de control	34
2.1. Controladora de vuelo	34
2.2. Ordenador de abordó	36
3. Sistemas de comunicación	37
3.1. Antena	37
3.2. Receptor	37
4. Sensores	39
4.1. GPS	39
5. Componentes adicionales	39
5.1. Dispositivo de visibilidad	39
5.2. Paracaídas	40
5.3. Cámara	40
 VII Prototipo	 42
1. Características	42
2. Conexiones	47
 VIII Entorno de programación	 50
1. Software	50
2. Sistema de comunicación	50
3. Estación de control en tierra	50
4. Lenguaje de programación	51

IX	Pruebas en tierra	52
1.	Simulaciones	52
1.1.	Misión con QGroundControl	53
1.2.	Despegue y aterrizaje reiterado	54
1.3.	Vuelo hasta unas coordenadas	56
1.4.	Cambio de modo por radio control	57
2.	Dispositivos de visibilidad	61
3.	Dispositivo de visión	63
X	Pruebas de vuelo	64
1.	Estructura	64
1.1.	Comprobaciones previas	64
1.2.	Vuelo con unidad de radio control	64
1.3.	Vuelo con QGroundControl	65
1.4.	Vuelo con el programa	65
2.	Vuelos	66
2.1.	Vuelo 1: Pixhawk 4	66
2.2.	Vuelo 2: Pixhawk 1	67
2.3.	Vuelo 3: Pixhawk 2 (Cube)	68
XI	Resultados	73
1.	Resumen económico	73
2.	Implicaciones ambientales	73
3.	Planificación y programación	74
4.	Conclusiones y trabajos futuros	74

Índice de figuras

1.	Amazon Prime Air	14
2.	DHL Parcelcopter 2.0	15
3.	DHL Parcelcopter 4.0	16
4.	Primer prototipo de Wing Project	16
5.	Vehículo de reparto y dron UPS	17
6.	Prototipo CAV	18
7.	Barómetro de mercurio	22
8.	Esquema de funcionamiento de un GPS	24
9.	Conexiones de la placa distribuidora	27
10.	Características motor Emax MT2216	31
11.	Motor Emax MT2216	32
12.	Hélice APC10x3.8	32
13.	Batería BrutePower 4S	33
14.	Regulador de velocidad ZTW Flash BLHeli-S de 30A	34
15.	Controladora Pixhawk 4	36
16.	Raspberry Pi 3 Modelo B	37
17.	Módulos de radio telemetría	37
18.	Receptor FS-IA6B	38
19.	Transmisor FS-I6X	38
20.	Kit Piwhawk 4	39
21.	Placa 4 LED RGB Matek Systems	40
22.	Raspberry Pi Módulo Cámara v2	41
23.	Configuración del prototipo	42
24.	Brazo del dron	43
25.	Montaje de los cuatro brazos	43
26.	Situación del GPS, antena y receptor	44
27.	Situación de la controladora, placa distribuidora y ordenador de abordo	45
28.	Situación de la batería	45
29.	Situación de las luces led	46
30.	Soporte de la cámara Raspberry	46
31.	Montaje final	47
32.	Esquema de conexiones	49
33.	Estación de control QGroundControl	51
34.	Logotipo Python	51
35.	Arquitectura de Ardupilot en SITL	52
36.	Simulación de quadcopter desde QGroundControl	53
37.	Esquema de misión en QGroundControl	53
38.	Simulación del programa de despegue y aterrizaje parte 1	54
39.	Simulación del programa de despegue y aterrizaje parte 2	55
40.	Simulación con mapa y consola	56
41.	Vehículo dirigiéndose a unas coordenadas	56
42.	Vehículo llega a su destino y aterriza	57
43.	Vehículo llega a casa y se completa la misión	57
44.	Conexión de la controladora y el receptor	58
45.	Simulación en Club Aeronáutico Egara	59
46.	Preparación de la unidad transmisora	59
47.	Ejecución de la simulación	60
48.	Interrupción de la simulación	60
49.	Vehículo cambia de modo y se detiene el programa	61
50.	Puesta a punto del ordenador Raspberry Pi	61
51.	Conexión de luces en Raspberry Pi	62

52.	Luces led encendidas	62
53.	Conexión del modulo cámara	63
54.	Primer vuelo hacia unas coordenadas	66
55.	Primera prueba finalizada	67
56.	Comportamiento inestable del prototipo	68
57.	Prototipo tercera prueba	68
58.	Misión con QGroundControl	69
59.	Comunicación con Raspberry en vuelo	70
60.	Despegue desde el programa	71
61.	Visión del vuelo desde el quadcopter	71
62.	Llegada al destino y aterrizaje autónomo	72
63.	Regreso al lugar de partida y fin de la misión	72
64.	Diagrama de Gant	74

Índice de tablas

1.	Características de los motores candidatos	30
2.	Características de la hélice elegida	32
3.	Características de la batería	33
4.	Características de los reguladores de velocidad preseleccionados	34
5.	Características receptor FS-IA6B	38
6.	Presupuesto final	73

Parte I

Introducción

1. Objeto

El objetivo de este trabajo es realizar el proyecto y diseño de un sistema de guiado para un RPA de servicio de transporte de mercancías puerta a puerta que cumpla con la normativa vigente en el Estado Español.

2. Alcance

A continuación se detalla una lista de los aspectos que se van a tratar en este trabajo. A lo largo del trabajo se van a desarrollar todos estos aspectos.

- Se realizará un estado del arte sobre la implementación de los drones en el transporte de mercancías y sus sistemas de guiado.
- Se hará un análisis de la normativa vigente en España.
- Se seleccionarán los sensores necesarios.
- Se seleccionarán los actuadores apropiados.
- Se diseñará la función de control.
- Se añadirán los sensores y actuadores seleccionados a un RPA ya diseñado y se programará con la función de control elegida.

3. Requisitos básicos

Los requisitos básicos que se han impuesto en este proyecto son:

- Se modificará un RPA del proyecto Aldora, que forma parte de UPC Space Program, para hacer el prototipo de la solución.
- El RPA debe ser capaz de transportar mercancías puerta a puerta.
- Tiene que cumplir con la normativa española.
- Peso máximo del paquete a transportar es de 1 kilogramo.
- La posición del RPA se controlará a partir de un GPS.

4. Justificación

La utilización de drones para el transporte de mercancías es sin duda una apuesta de futuro. Grandes compañías de transporte como Amazon ya están realizando pruebas con los primeros prototipos con la intención de implementar este sistema en los próximos años.

Este sistema de envío tiene múltiples ventajas. Una de ellas es la reducción del tráfico y por tanto de la contaminación, los drones son eléctricos y no necesitan quemar combustible para funcionar. Además, son capaces de llegar a lugares de difícil acceso donde los camiones o coches no pueden acceder. Otras de sus muchas ventajas es la posible reducción del tiempo de entrega, puesto que no hay barreras y siempre se seguirá la ruta más corta.

Como todo sistema de transporte también tiene sus problemas. A nivel tecnológico se necesita seguir trabajando sobretodo en la autonomía, ya que la mayoría de drones tienen una autonomía de poco más de media hora; y en la capacidad de carga, hasta ahora las pruebas que se han realizado han conseguido llevar paquetes con un peso máximo de 3 kilos [1].

Otro aspecto muy importante a tener en cuenta es la seguridad. Es necesaria una normativa que regule la actuación de los drones para preservar la seguridad de los ciudadanos y el espacio aéreo pero que no impida que estos puedan realizar su función. Por este motivo algunos países están haciendo cambios en sus normativas para adaptarlas a las necesidades actuales, como hizo España en 2017 [2].

Como conclusión, es un sistema de transporte que tiene muchas ventajas y puede ser muy beneficioso tanto para las empresas de transporte, para reducir gastos, como para el usuario por su rapidez, alcance y poca contaminación. El uso de drones para el transporte de mercancías es un proceso que se encuentra en pleno desarrollo y algunas compañías están trabajando en ello para que en poco tiempo sea una realidad ya que todavía hay algunos aspectos que se tienen que mejorar para que sea rentable y seguro.

Parte II

Estado del arte

Recientemente se ha empezado a desarrollar tecnología para utilizar drones con diferentes fines como por ejemplo la vigilancia, el transporte, la publicidad, etc. Numerosas empresas están trabajando para poder utilizar los drones en tareas cotidianas de forma que su aplicación resulte beneficiosa tanto para el usuario como para la empresa que las realiza. Centrándonos en el transporte, actualmente se trabaja en el transporte de mercancías, comida, correo postal, medicinas y ayuda humanitaria entre otros. Los objetivos que tienen los drones en estos transportes son varios: llegar a zonas de difícil acceso donde no se pueda acceder con otro medio de transporte o el trayecto sea de muy larga durada; reducir el tráfico en las ciudades, de esta forma también se contribuye a reducir la contaminación causada por los vehículos de transporte ya que en el caso de los drones los sistemas de propulsión son mayoritariamente eléctricos; también ayuda a reducir el tiempo de entrega, esto puede ser un factor clave en ciertos casos en que se necesite urgentemente algún producto.

Por lo que respecta al transporte de medicinas, se han llevado a cabo diversas pruebas orientadas sobre todo a abastecer áreas remotas o de difícil acceso. Un ejemplo de esto sería un prototipo de dron ambulancia creado por unos estudiantes holandeses, en 2014, que permitía el transporte de desfibriladores y establecía la comunicación en directo con un equipo de médicos que podían dar indicaciones sobre su uso y observar el estado del paciente. En el caso de la ayuda humanitaria, en 2016, la empresa Flirtey realizó la primera entrega totalmente autónoma de agua embotellada, alimentos de primera necesidad y un botiquín de primeros auxilios a una zona residencial deshabitada de Hawthorne, Nevada [3]. Ese mismo año, en Rwanda, el gobierno se asoció con la empresa Zipline International Inc. para construir un centro de distribución que utilizara drones para el transporte de sangre y productos farmacéuticos [4]. En 2017, Matternet y Swiss Post lanzaron la primera red de distribución de drones médicos de Suiza; ese mismo año, REMSA Health y Flirtey crearon un proyecto para enviar desfibriladores a cualquier persona que llamara al 911 desde el norte de Nevada describiendo los síntomas de un paro cardíaco [5].

Los drones también han sido utilizados para el transporte de comida. Empresas como Domino's Pizza o 7-Eleven han decidido apostar por este tipo de entrega como una apuesta de futuro [6]. Ambas empresas se asociaron con Flirtey en 2016 e hicieron las primeras pruebas con aviones no tripulados. En el ámbito de la hostelería, la cadena de hoteles Marriott International se asoció con DJI, en 2017, y utilizó drones en varios de sus hoteles para llevar cócteles y bebidas a las mesas de los huéspedes [7].

Por lo que respecta al servicio postal, en algunos países como Australia, Suiza, Alemania, Singapur o Ucrania se han realizado pruebas con drones para comprobar su viabilidad y la rentabilidad de este servicio [8].

El foco de este trabajo está puesto en el transporte de mercancías puerta a puerta, es decir en el uso que hacen de los drones las empresas de transporte que reparten paquetes desde un almacén central hasta el hogar del usuario. Los drones todavía no están plenamente adaptados en el sistema de transporte pero muchas de las grandes empresas del sector están trabajando para que su implementación se lleve a cabo lo antes posible. A continuación se detallan unas cuantas empresas de transporte y se explica los avances que han realizado así como los ensayos hechos hasta la fecha.

1. Amazon Prime Air

Amazon fue la primera empresa de transporte de mercancías que decidió apostar por los drones como medio de transporte. En 2013 se anunció desde la compañía que se empezaría a desarrollar tecnología para que el Amazon Prime Air fuera una realidad en cinco años vista. El objetivo era poder entregar paquetes de menos de 3 kilogramos (lo que supone entre un 80-90 % de las entregas que realiza Amazon) en menos de 30 minutos de forma completamente autónoma. A día de hoy todavía no se ha podido implementar pero se han hecho algunas pruebas con muy buenos resultados.

El 7 de diciembre del 2016 se realizó la primera entrega de forma completamente autónoma. Tuvo lugar en el Reino Unido en una zona de poca densidad de población donde había espacio suficiente para el aterrizaje. El dron realiza un despegue vertical con el paquete, lo transporta hasta el lugar de entrega, escanea la zona en busca de potenciales lugares donde aterrizar, aterriza verticalmente y deja el paquete suavemente en el suelo para volver a ganar altura de forma vertical. Este primer vuelo duró aproximadamente 13 minutos, el prototipo consiguió volar a una altura de hasta 100ft (30'5 metros) y es capaz de llevar paquetes de hasta 5 libras (2'27 kilogramos) [9].



Figura 1: Amazon Prime Air

Se han fabricado diversos modelos de dron y mecanismos de entrega y recogida de paquetes para ver cual es el que mejor se adapta a los diferentes entornos en que van a tener que trabajar. Algunos de los modelos son quadcopters, como el de la figura 1, pero también se han construido aviones de despegue vertical (VTOL). En el caso del modelo de avión puede volar a una altura de 400ft (122 metros) y tiene un alcance de hasta 15 millas (24 kilómetros). Este nuevo modelo también incorpora ciertas mejoras, por ejemplo, cuando el dron está a punto de llegar a su destino el usuario recibe un mensaje de aviso.

Con respecto a la seguridad, estos drones incorporan la tecnología "sense and avoid" que consiste detectar y esquivar cualquier obstáculo que puedan encontrar tanto en tierra como en el aire. Además, durante los vuelos de prueba se recopilan los datos para mejorar la seguridad y fiabilidad de los sistemas que incorporan. Otro aspecto relacionado con la seguridad y de gran importancia es la regulación del espacio aéreo. Se está trabajando en un sistema de gestión del tráfico aéreo que reconozca quién está volando qué avión no tripulado, a dónde se dirige y si se cumplen con los requisitos operativos.

Por el momento, los centros tecnológicos donde se están desarrollando estos prototipos están situados en Estados Unidos, el Reino Unido, Austria, Francia e Israel. Se trabaja en zonas de poca población donde el dron es capaz de descargar un paquete en un jardín o gran área sin obstáculos. En un futuro

se dará el salto a zonas más urbanizadas donde se deberán construir centros de aterrizaje, ya sea en azoteas, en estaciones locales o en vehículos de reparto cercanos.

2. DHL Parcelcopter

DHL es otra empresa de transporte que también está desarrollando tecnología para poder utilizar drones en el transporte de pequeños paquetes. Su objetivo principal es utilizar drones para llegar a áreas de difícil acceso, lugares donde el transporte por carretera no es posible o conlleva una gran cantidad de tiempo. DHL empezó a fabricar prototipos y realizar pruebas en diciembre de 2013 y hoy en día se encuentra en la cuarta generación.

El primer modelo, realizó su primer vuelo en diciembre de 2013, consistía en un quadcopter que podía transportar un paquete de pequeñas dimensiones. La prueba consistió en transportar un paquete de la orilla este del río Rin hasta la orilla oeste, donde se encuentra la sede de la corporativa DHL [10].



Figura 2: DHL Parcelcopter 2.0

Se implementaron una serie de mejoras a esta primera versión y en 2014 se realizó una misión en el mar del Norte -en la isla de Juist- con la ayuda del Parcelcopter 2.0. Este proyecto consistía en transportar medicinas y otros productos de primera necesidad. Estuvo realizando las entregas urgentes para una farmacia durante una semana. Fue la primera vez que se realizaban entregas a mar abierto en Europa con un dron completamente autónomo. Sus prestaciones mejoran considerablemente con respecto al primer modelo y además está diseñado para soportar las rachas de viento y condiciones meteorológicas propias de la zona del mar del Norte.

La tercera generación llegó en 2016, un avión con despegue vertical con las alas capaces de rotar por los puntos del encastre con el fuselaje, de esta forma las hélices y las alas cambian de orientación según la fase de vuelo. Después de tres meses de pruebas este dron se integró en el sistema de entrega de paquetes de la empresa. Para ello también idearon un sistema de carga y descarga de paquetes totalmente automatizado denominado "Parcelcopter Skyport". Realizaba entregas desde la estación de esquí de Reit im Winkl hasta Winklmoosalm, el tiempo aproximado desde el despegue hasta el aterrizaje del dron es de 8 minutos, este mismo trayecto en coche conlleva unos 30 minutos.

La cuarta y, de momento, última generación del Parcelcopter consiste en un avión completamente autónomo que en 2018 realizó una misión de ayuda humanitaria. Durante seis meses entregó de forma autónoma medicamentos a una isla en el Lago Victoria, el alcance de los vuelos era de 60 kilómetros y el tiempo empleado de aproximadamente 40 minutos. En total unos 2.000 minutos de vuelo y 2.200 kilómetros recorridos [11].



Figura 3: DHL Parcelcopter 4.0

Para más información sobre las propiedades de cada uno de los prototipos (distancia volada, autonomía, velocidad, etc.) en el anexo A se adjunta la hoja de características que DHL pone a la disposición de los usuarios.

3. GoogleX Project Wing

Google también está desarrollando tecnología dron para el transporte de mercancías. En agosto del 2014 anunciaron que llevaban dos años haciendo pruebas con este tipo de vehículos en Australia a través del programa GoogleX conocido como Project Wing. Project Wing está trabajando para crear un servicio de transporte autónomo con vehículos no tripulados con el objetivo de llevar mercancías a lugares de difícil acceso y reducir el tráfico en las ciudades, de esta forma también se podrá reducir la contaminación que genera el sector del transporte. Además, están desarrollando tecnología para gestionar el tráfico de vehículos no tripulados que permita que estos puedan volar alrededor de otros vehículos no tripulados, aviones tripulados y cualquier otro tipo de obstáculos como árboles, edificios o cables de la luz.

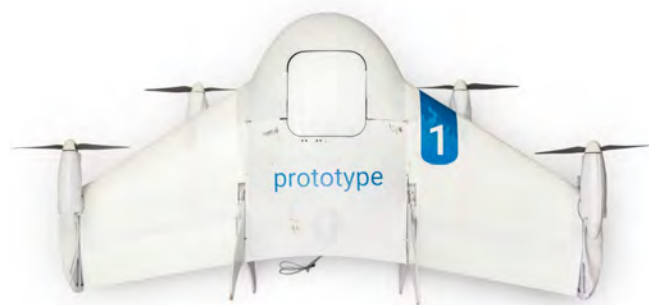


Figura 4: Primer prototipo de Wing Project

El diseño del primer prototipo consiste en un dron capaz de alcanzar los 120 km/h, propulsado por motores eléctricos y por tanto no genera emisiones. Puede volar hasta 120 de altura y entregar un paquete en una ubicación con margen de error del tamaño de una puerta. Los algoritmos de aprendizaje autónomo ayudan al dron a encontrar un lugar seguro y adecuado para realizar la entrega a pesar de los obstáculos que pueda encontrar [12].

En el año 2012 se empezó a desarrollar el Project Wing con el objetivo de estudiar como entregar con vehículos autónomos todo tipo de objetos de forma rápida y segura. Empezaron a diseñar un sistema que pudiera transportar desfibriladores a personas que sufrieran un ataque al corazón con la

esperanza de poder salvar más vidas si los drones eran capaces de llegar más rápido por el aire que las ambulancias por carretera. Pero esta era una tarea demasiado compleja y decidieron centrarse en diseñar un sistema de transporte de paquetes pequeños en situaciones cotidianas donde la velocidad en la entrega fuera un factor clave. En 2013, se diseñaron y construyeron diferentes prototipos y se realizaron los primeros vuelos de prueba. No fue hasta 2014 que se llevó a cabo la primera entrega real, esta entrega se realizó en Queensland, una zona rural de Australia. Consistió en entregar un botiquín, golosinas, comida para perros y agua para agricultores. En setiembre de 2016 se entregaron burritos a los estudiantes de Virginia Tech, fue la entrega de mayor duración y distancia que se había llevado a cabo en Estados Unidos hasta la fecha.

Actualmente se están trabajando en entregas en zonas suburbanas de Australia. El objetivo de estas entregas es estudiar el modo de encontrar la mejor ruta hacia un punto y el lugar más adecuado para realizar la entrega.

4. UPS

Esta empresa utiliza un concepto un poco diferente respecto a las que se han analizado anteriormente. En este caso, el objetivo es usar drones para realizar entregas de paquetes en zonas residenciales rurales, pero la carga del paquete en vez de hacerse desde una sede central se hace desde una furgoneta o camión de reparto eléctrico.

La primera prueba de este sistema se realizó en febrero del 2017. Tanto el vehículo de reparto como el dron utilizados han sido fabricados por la empresa Workhorse. El sistema de reparto consiste en el despegue del dron cargado desde el techo del camión, el vuelo autónomo hasta el punto de entrega, donde deja el paquete, y finalmente regresa a la ubicación donde se encuentre el vehículo de reparto en ese momento. Mientras el dron realiza la entrega, el conductor continúa la ruta y realiza otra entrega. Este método de entregas está pensado para zonas rurales donde las casas se encuentran muy lejos unas de otras de forma que el conductor puede ahorrar mucho tiempo y además reducir las emisiones producidas por el vehículo de reparto [13].

Respecto al dron utilizado en las pruebas, se trata de un octocopter (dispone de ocho motores), su autonomía es de aproximadamente 30 minutos y puede transportar una carga máxima de hasta 4 kilos y medio. Los drones disponen de una caja donde almacenar el paquete en la parte inferior. Cuando están en el techo del vehículo de reparto se abre una escotilla de forma que desde dentro del camión se tiene acceso a la caja que incorpora el dron. El conductor del camión introduce el paquete a entregar y, cuando está listo, desde una tablet da la orden para que el dron despegue y se dirija de forma autónoma a la dirección preestablecida. Además de realizar la carga del paquete desde el vehículo de reparto, cuando los drones están en él también cargan sus baterías.



Figura 5: Vehículo de reparto y dron UPS

El CEO de Workhorse, Stephen Burns, comentó que el siguiente paso es conseguir llevar entre tres y cinco drones en el techo de un vehículo de reparto de forma que este se convertiría prácticamente en un almacén móvil desde el cual los drones despegarían y aterrizarían de forma continua.

En una línea muy parecida a la de Workhorse, Mercedes-Benz está trabajando en una furgoneta eléctrica, llamada Vision Van. Este vehículo de reparto es capaz de transportar dos drones en el techo y cargar la mercancía en ellos de forma totalmente automatizada. Incluye innovadores algoritmos que controlan la preparación de los pedidos, la carga de paquetes, la gestión totalmente automatizada del espacio de carga, además de la planificación de las rutas para el propio vehículo de reparto y las de los drones cuando deben realizar una entrega. Los quadcopters de los que dispone pueden transportar una carga máxima de 2 kilogramos y tienen un alcance de 10 kilómetros de radio [14].

5. Boeing CAV

A mediados de 2018 Boeing presentó un prototipo del Cargo Air Vehicle (CAV), un dron capaz de transportar una carga útil de 500 libras (unos 225 kg). Es un vehículo aéreo no tripulado completamente eléctrico y con capacidad de aterrizaje y despegue vertical [15].

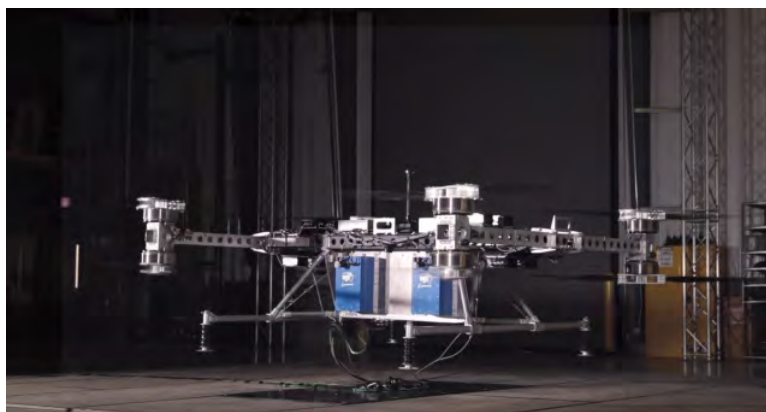


Figura 6: Prototipo CAV

Según la compañía, este dron fue diseñado y construido por un equipo de ingenieros de Boeing en tan solo tres meses. El CAV dispone de ocho rotores, pesa 339 kilogramos y mide 4'57 metros de largo por 5'49 de ancho. En el mes de julio se realizaron las primeras pruebas de vuelo en un laboratorio de investigación de Boeing, de las cuales se conocen muy pocos datos [16].

Parte III

Análisis de la normativa

En este apartado se va a analizar el Real Decreto 1036/2017, que entró en vigor el 29 de Diciembre de 2017 y que *regula la utilización civil de las aeronaves pilotadas por control remoto*. En ella se indica que *corresponde a los Estados velar porque el vuelo de estas aeronaves sin piloto en las regiones abiertas al vuelo de aeronaves civiles se regule de tal modo que les evite todo peligro*. Como consecuencia de los avances tecnológicos sumados a la reducción de los precios de adquisición estos vehículos no tripulados, han hecho que se introduzcan en el espacio aéreo de forma masiva y esto puede comportar riesgos para la seguridad. Es necesario, por tanto, establecer un régimen jurídico para poder satisfacer la necesidades de la industria y a la vez garantizar la seguridad tanto de las operaciones como de los ciudadanos.

En consecuencia, *este real decreto establece el marco jurídico definitivo aplicable a la utilización civil de las aeronaves pilotadas por control remoto (RPA) de masa máxima de despegue inferior a los 150 kg*. También subraya que este real decreto *no regula el uso de aeronaves civiles no tripuladas que no permiten la intervención del piloto en la gestión del vuelo*, conocidas como aeronaves autónomas y que su uso no está permitido en el espacio aéreo español ni en el territorios donde España es el responsable de los servicios de tránsito aéreo.

El RPA que se pretende diseñar en este proyecto tendrá una masa máxima de despegue de 2 kg y el objetivo es destinarlo al transporte de mercancías, es decir que pueda transportar paquetes de un punto a otro. Según este real decreto, las aeronaves pilotadas por control remoto pueden utilizarse para:

- *La realización de trabajos técnicos o científicos y vuelos de prueba de producción y mantenimiento, de demostración, investigación y desarrollo de nuevos productos o para demostrar la seguridad de las operaciones específicas de trabajos técnicos o científicos.*
- *No contempla la posibilidad de autorización de otras operaciones con aeronaves civiles pilotadas por control remoto, señaladamente el transporte. Estas operaciones con aeronaves pilotadas por control remoto quedan diferidas a un posterior desarrollo reglamentario, ya que no existen a fecha de hoy condiciones objetivas de seguridad para su autorización.*

Por lo tanto, actualmente en el territorio español no es posible realizar operaciones de transporte de personas ni de mercancías. El dron diseñado por lo tanto no podrá transportar ningún paquete, aún y así para la probar su sistema de guiado se seguirá analizando la normativa para cumplir con todas sus especificaciones. En el anexo B se hace un resumen de los puntos que afectan al tipo de dron que se pretende diseñar.

1. Requisitos de la aeronave y los sistemas

No es necesaria la inscripción en el Registro de Matrícula de Aeronaves Civiles ni tampoco la obtención del certificado de aeronavegabilidad. Deberá llevar en su estructura una placa identificativa ignífuga con el nombre del propietario y los datos para ponerse en contacto con él.

El piloto ha de poder intervenir en cualquier momento en el control de la aeronave, siendo el responsable de evitar colisiones. Un piloto no puede estar controlando más de un RPA al mismo tiempo. Si la aeronave pesa menos de 2 kg, el mantenimiento lo puede hacer el propio operador siguiendo las instrucciones del fabricante.

Es necesario un sistema de comunicación capaz de sostener comunicaciones bidireccionales con la estación de control remoto en las frecuencias pertinentes. Es deberán incorporar sistemas para controlar la posición del aeronave en todo momento y controlar que no salga del aérea de vuelo limitada. Además se deben incorporar luces u otros dispositivos para asegurar la visibilidad.

En caso de realizar un vuelo sobre edificios, poblaciones o personas al aire libre será necesario designar zonas de aterrizaje de emergencia y disponer de dispositivos que reduzcan la energía en caso de caída.

Si la aeronave vuela fuera del alcance visual del piloto se deberá incorporar un dispositivo de visión orientado hacia delante.

2. Condiciones de vuelo

Las operaciones deberán realizarse de día y en condiciones meteorológicas de vuelo visual, aunque hay la posibilidad de realizar vuelos nocturnos si se desarrolla un estudio de seguridad y se solicita a la Agencia Estatal de seguridad Aérea (AESA).

Para volar en una zona donde no hay edificios, poblaciones ni personas al aire libre se deberá volar dentro del alcance visual del piloto u observadores (500 m) y a una altura máxima de 120 metros respecto el suelo u obstáculo más elevado. Si la aeronave pesa menos de 2 kg se podrá volar fuera del alcance visual del piloto y dentro de la zona de emisión de radio de la estación de control remoto, pero antes se deberá informar por medio de un NOTAM (para que el resto de usuarios tengan constancia) y detallar si se dispone de sistemas que permitan al piloto detectar y evitar a otros usuarios.

Si se vuela sobre edificios, poblaciones o personas al aire libre se ha de pedir una autorización previa a la AESA. Se deberá volar dentro del alcance visual del piloto (100 m) y guardar una distancia de seguridad mínima de 50 metros respecto a estructuras y personas y a una altura máxima de 120 metros respecto el suelo u obstáculo más elevado.

Se puede volar en un espacio aéreo controlado siempre dentro del alcance visual del piloto y con autorización previa de la AESA.

3. Pilotos y observadores

Los pilotos, en el caso profesional, necesitan una licencia de piloto que consiste en una acreditación teórica y práctica. Para mantener la acreditación es obligatorio haber volado como mínimo 3 veces en los últimos 3 meses en cada categoría de aeronave (los entrenamientos en simulador también cuentan). En cuanto a los observadores solo es necesario que tengan conocimientos teóricos sobre los RPA.

No se podrá pilotar un RPA desde un vehículo en movimiento a menos que se planifique la ruta con antelación y se asegure que no se interpondrá ningún obstáculo en dicha ruta.

Parte IV

Sensores

En este apartado se van a detallar todos los sensores necesarios en un RPA así como sus características y funciones principales. Los sensores son muy importantes ya que su función principal va a ser adquirir información para que después esta sea procesada y analizada, en este caso, por la controladora de vuelo. Esto va a permitir que el dron pueda maniobrar o corregir su posición en función de las lecturas recibidas a través de los sensores. Los sensores que son imprescindibles para el vuelo, son los siguientes:

- Acelerómetro
- Giroscopio
- Barómetro
- Magnetómetro
- GPS

1. Acelerómetro

El acelerómetro o también denominado sensor de aceleración se usa para medir la vibración o aceleración a la que un objeto está sometido. Estos sensores convierten la variación de aceleración en una señal eléctrica proporcional a la magnitud de la señal física recibida. Existen diversos tipos de acelerómetros. Los principales son los capacitivos, piezoeléctricos y piezoresistivos [17].

Una propiedad muy importante al escoger un acelerómetro es su rango de aceleración, es decir, el rango de Gs que es capaz de medir. Disponen de una banda de paso en la que se muestra la aceleración mínima y máxima que es capaz de medir. También es muy importante la sensibilidad, suele estar limitada por la precisión de bit que tenga el convertidor analógico-digital que lleva integrado. Otra de sus características es la frecuencia de muestreo máxima, esta propiedad determinará la precisión con la que el acelerómetro puede medir la aceleración instantánea o el cambio de velocidad.

Se utilizan en un gran rango de aplicaciones: dinámica de vehículos, detección de la orientación de un teléfono móvil, estabilidad de imagen, detección de choques y dispositivos antirobo.

En el caso de los drones suele utilizarse acelerómetros de tres ejes, mide la aceleración estática en el eje vertical y la dinámica en el eje horizontal en un plano XY. Se usa para determinar la posición y orientación respecto a la superficie de la tierra que tiene el dron durante el vuelo. La tecnología que se utiliza es muy similar a la de un giroscopio y se denomina MEMS (Micro Electro-Mechanical Systems).

2. Giroscopio

Los giroscopios, o también denominados sensores de velocidad angular, son unos dispositivos que detectan los cambios en la velocidad angular, es decir, es capaz de medir la velocidad de rotación sobre un eje. Los giroscopios son sensibles al movimiento rotatorio y a los cambios de orientación de los objetos [18].

Los giroscopios tienen principalmente estas tres aplicaciones:

- Detección de la velocidad angular: detectan el valor de la velocidad angular que tiene un objeto. Esto se usa para medir la cantidad de movimiento angular. Un ámbito de aplicación sería el deporte.
- Detección de ángulos: detectan la velocidad angular producida por un movimiento, este dato es enviado a una CPU encargada de realizar el cálculo del ángulo recorrido en el movimiento. Esto se aplica en los sistemas de navegación de coches, en teléfonos móviles o en las controladoras de los videojuegos.
- Mecanismos de control: detectan la vibración producida por un factor externo, convierten esta entrada física en una señal eléctrica y la envían a una CPU. Se utiliza para corregir la orientación o el equilibrio de un objeto. Se aplica en cámaras con corrección por movimiento o en el control de vehículos.

En el caso de los drones, el giroscopio se encarga medir la orientación para poder compensarla. Durante el vuelo los drones pueden sufrir el efecto de las ráfagas de viento y otras perturbaciones que afecten a su orientación (ángulos de balance, cabeceo y guiñada), el giroscopio detecta estos cambios y con la ayuda de una controladora se pueden corregir. Normalmente el giroscopio y el acelerómetro forman parte de un mismo componente.

3. Barómetro

El barómetro es un sensor que mide la presión atmosférica, la unidad de medida es el hectopascal(hPa). Existen principalmente dos tipos de barómetro según su principio de funcionamiento: de mercurio o aneroide [19].

- Mercurio: su funcionamiento se basa en un tubo con mercurio en su interior, según la presión exterior el mercurio se desplazará por dentro del tubo. Si la presión atmosférica aumenta, el aire ejercerá más fuerza sobre el mercurio cosa que hará que este suba por el tubo. Por el contrario, si la presión atmosférica disminuye descenderá el nivel de mercurio del interior del tubo.

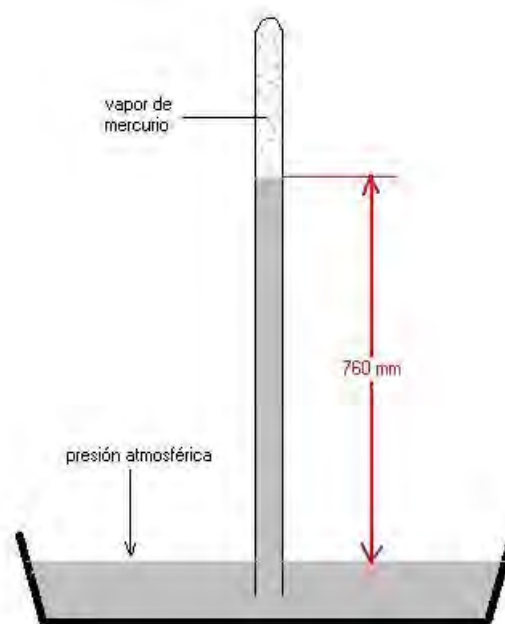


Figura 7: Barómetro de mercurio

- Aneroide: consiste en un instrumento circular con una aguja que gira señalando la presión que está graduada en el anillo exterior. En el interior de este tipo de barómetro hay una cápsula en que se ha generado el vacío. Esta cápsula contiene elementos elásticos que se deformaran según los cambios de presión que se produzcan en el exterior. Como consecuencia de estas deformaciones se modificará la posición de una aguja.

El barómetro utilizado en los drones se del tipo aneroides y también se suele denominar altímetro ya que su principal función es medir la altura a la que se encuentra el dron a partir de la presión atmosférica medida. Generalmente viene incorporado de serie en todas las controladoras de vuelo para poder medir la altitud, en caso de incorporar un GPS los datos se cogerán de este último. Aún así es importante incorporar un barómetro por razones de seguridad, si el GPS fallará no se tendría ninguna referencia de la altitud de vuelo y eso podría ser peligroso ya que existe un gran riesgo de perder el control del dron.

4. Magnetómetro

El magnetómetro es un sensor capaz de medir la intensidad del campo magnético y, en algunos casos, también su dirección. Están divididos en dos grandes grupos: escalares, solo pueden medir la fuerza del campo magnético; vectoriales, además de medir la fuerza también señalan su dirección. Se usan para aplicaciones muy diversas, desde la detección de vetas de minerales magnéticos y depósitos de hierro, hasta la detección de submarinos. Pero el uso más conocido es el de las brújulas. Los magnetómetros suelen ir integrados en los circuitos de muchos aparatos electrónicos como teléfonos móviles o tabletas.

Un magnetómetro mide la densidad de flujo magnético en el punto del espacio donde se encuentra el sensor. La densidad de flujo magnético que hay en el aire es directamente proporcional a la intensidad del campo magnético. Los materiales magnéticos crean distorsiones en el flujo magnético que hay a su alrededor y los magnetómetros son capaces de medir estas distorsiones [20].

En un dron, el magnetómetro se usa como brújula y su función es medir en todo momento su orientación respecto a los puntos cardinales, es decir, conocer siempre donde se encuentra el norte. Hay que vigilar a la hora de situar el magnetómetro en el dron ya que el campo magnético se puede ver afectado por la intensidad de la corriente que circula por los cables de distribución, batería y reguladores de velocidad. La mejor solución es ponerlo en un sitio elevado y lejos de estos componentes.

5. GPS

GPS son las siglas de Sistema de Posicionamiento Global. Este sensor se encarga de medir la latitud, longitud y altura en la que se encuentra este objeto en todo momento. Su funcionamiento se basa en el concepto de la triangulación. Están integrados en un gran número componentes como teléfonos móviles, relojes, sistemas de navegación, etc.

Alrededor de la Tierra orbitan más de 30 satélites a una altura de 20.000 kilómetros, de forma que en cualquier lugar del planeta como mínimo cuatro de ellos van a ser visibles, es decir, el GPS podrá recibir su señal. Estos satélites envían una señal con su posición en tiempo real cada ciertos intervalos de tiempo, dichas señales se propagan a la velocidad de la luz formando unas esferas en el centro de las cuales se encuentra el satélite. El GPS actúa como receptor y es capaz de captar estas señales y calcular a que distancia se encuentra del satélite emisor, sabiendo el tiempo que ha tardado en recibir el mensaje [21].

Como se ha mencionado anteriormente, los sistemas GPS se basan en el principio de la triangulación. Si un receptor recibe señal del satélite A, podrá calcular a que distancia se encuentra de él. En el caso de la figura 8 se puede deducir que se encontrará ubicado en algún punto del círculo azul. Se

necesita como mínimo conocer la posición de dos satélites más para poder determinar la ubicación exacta del receptor. En este caso se encontrará en la intersección de los tres círculos, o en realidad esferas. La precisión de la ubicación será más exacta cuantos más satélites pueda recibir señal.

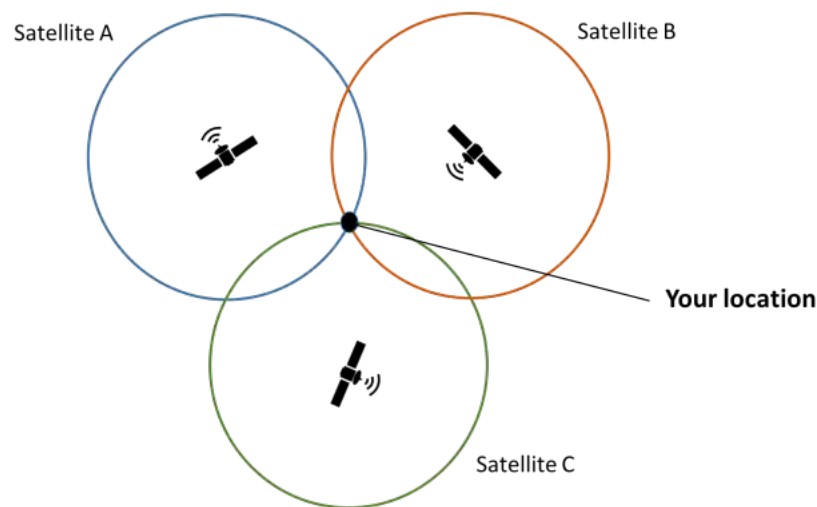


Figura 8: Esquema de funcionamiento de un GPS

Este receptor es muy importante en el caso de los drones, especialmente si vuelan de forma autónoma, puesto que gracias a su conexión con la controladora puede guardar las coordenadas de los puntos por donde tiene que pasar así como el lugar al que debe regresar. Es muy útil esta última opción ya que en caso que la batería se esté agotando o que surja algún problema el dron es capaz de volver al lugar desde donde despegó. También es posible obtener otros datos como la velocidad a la que se está desplazando el vehículo. Esto nos ayuda a obtener una telemetría completa sobre el estado del dron en todo momento.

Parte V

Actuadores

En este apartado se van a explicar los actuadores que es imprescindible incorporar en un RPA para que este sea capaz de volar de forma estable y controlada. Los actuadores son un conjunto de componentes y cada uno de ellos es una pieza clave para el correcto funcionamiento del dron. Se van a analizar todos estos componentes explicando su función principal dentro del conjunto y sus características más importantes que hay que tener en cuenta en el dimensionado o a la hora de adquirir un dron. A continuación se muestra una lista con los actuadores necesarios en un dron:

- Motor y hélice
- Batería
- Regulador de velocidad (ESC)
- Controladora de vuelo
- Placa distribuidora (PDB)
- Ordenador de abordo
- Sistema de radio control
- Telemetría

1. Motor y hélice

En los drones el conjunto motor-hélice es el que genera la tracción necesaria para que el dron pueda volar. Existen diferentes tipos de drones y configuraciones en función del número de motores usados. Generalmente se dividen en tricopter (3 motores), quadcopter (4 motores), hexácopter (6) y octocopter (8).

Los motores se encargan de transformar la energía eléctrica recibida en un movimiento circular que hace girar las hélices. Al girar, las hélices generan sustentación y eso hace que el dron se levante. Hay dos tipos de motores principalmente: bifásicos o con escobillas y trifásicos o sin escobillas. Los bifásicos suelen usarse en drones pequeños y de menor coste mientras que los trifásicos se usan cuando se necesita mayor potencia y control. Como se detallará más adelante en el apartado de elección de componentes, las características principales de los motores son: la tracción que son capaces de generar, dependerá del peso que sea necesario levantar; la velocidad de giro, se mide en kV (revoluciones por voltio); y el voltaje necesario para funcionar, será necesaria una batería capaz de subministrar este voltaje a todos los motores.

Las hélices suelen ser principalmente de dos o tres palas, las de tres palas sirven para dotar al dron de mayor estabilidad pero consumen más energía. Los materiales de fabricación más comunes son el plástico, la fibra de carbono o el nylon. Las principales características son el diámetro, a menudo viene fijado o recomendado en función del motor elegido; y el paso, el ángulo de inclinación de las palas.

2. Batería

La batería es la fuente de alimentación del dron. Como se ha comentado en el estado del arte, una de las principales ventajas del dron respecto a otros vehículos de transporte es que se propulsa con un sistema eléctrico y por tanto no contaminan. Generalmente, las baterías están compuestas de polímeros de litio, por eso en ocasiones se pueden encontrar denominadas como LiPo. Además la batería suele usarse como elemento estabilizador, ya que de su posición dependerá mayormente el centro de gravedad del dron, puesto que en general es el elemento de mayor peso.

La principal característica de las baterías es el numero de celdas. Se pueden encontrar baterías que van de 1S a 6S donde S corresponde al numero de celdas conectadas en serie, cada celda genera 3'7 voltios; de esta forma podemos calcular el voltaje nominal total que genera la batería. El numero de celdas vendrá determinado por el voltaje que necesiten los motores. Cuanto mayor sea el numero de celdas, mayor será la tensión y se reducirá la corriente necesaria, pero por contra también aumentará el peso de la batería. Otra característica muy importante es el ratio de descarga, se suele expresar en Cs. Es una medida del tiempo que tarda en descargarse una batería. Por ejemplo, una batería de 25C, trabajando en máximas condiciones, tardaría 1/25 horas en descargarse, es decir, 2'4 minutos. Por último también es necesario conocer la capacidad, es decir, los Amperios por hora que la batería es capaz de suministrar. Dados el ratio de descarga y la capacidad se puede calcular la máxima corriente que la batería es capaz de suministrar, el resultado se obtiene de multiplicar ambos factores.

3. Regulador de velocidad

Los reguladores de velocidad, también denominados ESC (Electronic Speed Control) o variadores, se encargan de suministrar la corriente necesaria a los motores en todo momento. Por tanto, sus funciones principales serán, transformar la corriente continua procedente de la batería en corriente alterna para los motores trifásicos o sin escobillas; y regular la velocidad de giro del motor.

Las características principales de este componente son su amperaje y tensión máxima. El amperaje va a depender el amperaje máximo del motor al que irá conectado. Como se detallará en el apartado de elección de componentes, es importante contemplar un margen de seguridad a la hora de dimensionar los reguladores de velocidad, puesto que son elementos muy sensibles y un pico de corriente podría quemarlos. Además, es necesario que pueda soportar la tensión de la batería a la que también va conectado. Por último, hay algunos reguladores de velocidad con mayor frecuencia de actualización, es decir, que el numero de veces por segundo que se comprueban las instrucciones de la controladora y ajustan la velocidad es mayor. Cuanto mayor es la frecuencia de actualización mayor es la estabilidad y se consigue mayor control.

4. Placa distribuidora

Una placa distribuidora, o también conocida por las siglas PDB (Power Distribution Board), se encarga de distribuir la corriente. La batería se conecta a la placa distribuidora y esta a su vez va conectada a los reguladores de velocidad a los cuales debe suministrar corriente. La placa puede ir integrada en el chasis del dron o ser un componente a parte. En caso de ir integrada, el chasis no puede estar compuesto de fibra de carbono ya que es un material que conduce la electricidad. En la siguiente imagen se muestra como sería la conexión de la batería y los reguladores en la placa distribuidora, en este caso se usan cuatro reguladores lo que indica que se trata de un quadcopter [22].

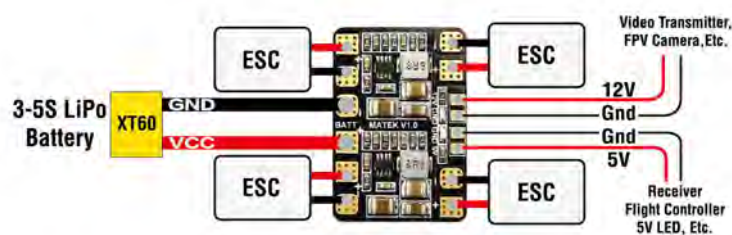


Figura 9: Conexiones de la placa distribuidora

Además de distribuir la corriente, también es capaz de regular la tensión, el regulador de voltaje se conoce como BEC. Por ejemplo, si la batería utilizada es una 4S, proporciona 14'8 voltios, los reguladores conectados a ella necesitan ese voltaje para funcionar pero es posible que se quieran conectar otros sensores o actuadores que funcionen con otro voltaje. El regulador de voltaje es un circuito que convierte los 14'8 voltios de entrada en 5 o 12 voltios, normalmente, de corriente continua. Existen diferentes tipos de placas con una sola salida de 5 y 12 voltios, como la de la figura 9, con doble salida BEC y algunas a además de incorporar BEC también llevan filtros LC. Estas últimas se usan para señales de vídeo y el filtro tiene la función de evitar el ruido estabilizando la corriente suministrada al sistema de vídeo.

La principal característica que debe cumplir la placa distribuidora es soportar el corriente máximo que puede circular por los reguladores de velocidad. Para ello, se deberá comprobar en la ficha de características el valor de esa corriente y multiplicarlo por el numero de reguladores que estén conectados. Generalmente no se alcanza ese máximo de corriente ya que como se ha explicado anteriormente los reguladores se sobredimensionan para que no se quemen si se produce un pico de corriente. También es importante conocer el tipo de dron en el que se va a utilizar, es decir, para un quadcopter será necesario conectar cuatro reguladores y por tanto son necesarias cuatro salidas; en el caso de un hexacopter se necesitarán seis salidas para los seis reguladores; y en un octacopter, ocho. Por último, se deberá tener en cuenta los equipos auxiliares que se quieren incorporar al dron para elegir el tipo de placa; con filtros o sin y con simple o dobles salida BEC.

5. Controladora de vuelo

La placa controladora de vuelo es la encargada de mantener el dron volando de forma estable. Normalmente lleva incorporados una serie de sensores con cuyas lecturas es capaz de recopilar la información suficiente para mantener la estabilidad. Los sensores que habitualmente llevan incorporados las controladoras son: el acelerómetro y el giroscopio, un magnetómetro, un barómetro o altímetro y en algunas ocasiones un GPS. Además de los sensores también dispone de un procesador capaz de transformar las lecturas en ordenes en segundos.

La controladora recibe las lecturas de los sensores que lleva incorporados y de los sensores externos que se deseen conectar y se encarga de realizar las actuaciones pertinentes en función de dichas lecturas para mantener un vuelo estable. También recibe las ordenes dadas desde la unidad de control remoto en tierra (o mando), las procesa y transmite ordenes para que los motores giren a la velocidad necesaria con el fin de realizar la actuación deseada por el piloto sin comprometer la estabilidad del dron. En el caso de los drones pilotados de forma autónoma, los datos no los recibirá del mando de control sino desde el ordenador que lleve abordo y que previamente haya sido programado para realizar una serie de operaciones.

6. Ordenador de abordó

Para que un dron sea capaz de volar de forma autónoma es necesario programarle una serie de ordenes o instrucciones que debe cumplir durante el vuelo. El ordenador de abordó es el componente que realiza esta función. Este ordenador consiste en una pequeña placa, del tamaño de una tarjeta, que puede ser programada utilizando el lenguaje de programación y el entorno apropiado. De esta forma es posible introducirle rutas, coordenadas GPS por las que deberá pasar el dron, funciones para despegar y aterrizar o cualquier otra funcionalidad que nos interese que lleve a cabo el vehículo.

El ordenador de abordó va conectado a la controladora de vuelo, de forma que las instrucciones previamente programadas se envían a la controladora y esta realiza los cambios necesarios para que el dron las ejecute de manera correcta y volando en todo momento de forma estable. De esta forma el ordenador de abordó sustituye la función del mando o unidad de control remoto.

7. Sistema de radio control

Para poder controlar el dron desde tierra se necesita un transmisor y un receptor. El transmisor, en este caso, será el mando que transformará los movimientos de los joystick o gimbals en señales y estas señales serán enviadas por ondas de radio. El receptor es una unidad instalada en el dron que recibe estas ondas. Se conecta a la controladora de vuelo de forma que las ordenes dadas por el piloto en tierra se comunicarán a la controladora y esta realizará las operaciones oportunas.

La frecuencia de radio utilizada para esta comunicación es normalmente 2'4 GHz aunque para drones de larga distancia pueden utilizarse otras frecuencias, como por ejemplo 433 MHz, 900 MHz y 1'3 GHz. La unidad de radio control dispone de un software que se encarga de escanear la frecuencia óptima y detectar interferencias, este protocolo se realiza cientos de veces cada segundo y de esta forma se evitan los lapsos de transmisión que ocurrían con los sistemas de radio control antiguos [23].

Una de las características más importantes es el número de canales. Cada canal permite el movimiento una acción, el mínimo de canales necesarios para poder controlar un dron es cuatro, eso permite realizar los movimientos de cabeceo, guiñada y alabeo y el cuarto sirve para acelerar. Normalmente, se usan canales adicionales, denominados auxiliares, y en el mando serían accionados mediante botones o interruptores para activar algún sistema o cambiar el modo de vuelo.

Es importante tener en cuenta la compatibilidad entre el transmisor y el receptor. Generalmente los transmisores solo funcionan con receptores de la misma marca aunque si disponen de una tarjeta multiprotocolo pueden funcionar con receptores de otras marcas. Para que puedan comunicarse es necesario realizar un protocolo denominado emparejamiento. Este protocolo solo se realiza la primera vez y se puede emparejar un mismo transmisor con distintos receptores.

8. Telemetría

Para poder establecer comunicación con el dron desde tierra es necesario incorporar receptores y transmisores. Si queremos recibir datos sobre la velocidad, ubicación o cualquier información que nos puedan proporcionar los sensores que lleve incorporados, será necesario una unidad emisora que sea capaz de enviar esta información a la unidad receptora en tierra. La unidad receptora suele ser un ordenador.

Generalmente, este sistema de comunicación se basa en una antena conectada a la controladora de vuelo del dron y otra antena receptora conectada a un ordenador en tierra que funciona como estación de control. Esta comunicación permite recibir datos en tiempo real lo que hace posible monitorizar el estado del dron, ajustar algún parámetro en caso de que sea necesario o conocer el modo de vuelo entre otras opciones.

Existen diversos sistemas de telemetría, principalmente, por radio, wifi o bluetooth [24]. La elección del sistema utilizado dependerá en mayor medida de la controladora de vuelo elegida, ya que quizás no sea compatible con todas las opciones. Otro factor determinante es la distancia, puesto que no tienen el mismo rango de conexión. La radio telemetría envía los datos por señales de radio, se necesita una antena para enviar los datos y otra para recibirlos; su rango de conexión es de unos 300 metros. La telemetría wifi no necesita antena receptora ya que si el ordenador dispone de wifi puede recibir los datos el propio ordenador; generalmente, el rango de conexión es más corto que el de las radios pero la velocidad de transmisión de datos es más alta. Finalmente, la telemetría bluetooth no necesita de una antena sino de un modulo bluetooth que se conectará con el ordenador de tierra. Este último es el sistema de telemetría con un rango de transmisión más corto, 50 metros.

Parte VI

Elección de componentes

En este apartado se van a elegir los componentes que formarán parte del dron con el que se realizarán las pruebas de vuelo. Estos componentes son los explicados en los apartados previos de sensores y actuadores. Primero se hará la selección de los actuadores y posteriormente se añadirán los sensores necesarios. Como se ha explicado anteriormente, este dron pertenece al proyecto Aldora, que forma parte UPC Space Program.

1. Sistema motor-propulsor

La función principal de este sistema es poder levantar el dron del suelo, esta formado por los propios motores, hélices, batería y reguladores de velocidad. Además, se encarga de proporcionar la energía suficiente para alimentar todos los componentes que forman parte del dron.

1.1. Motor

El primer componente que hay que elegir es el motor, de él va a depender la elección de la batería y los reguladores de velocidad. Para elegir un motor primero hay que decidir que configuración se va a realizar (tricopter, quadcopter, hexacopter, octacopter). En este caso se va a optar por la construcción de un quadcopter puesto que permite mayor capacidad de carga que un tricopter y, por otra parte, es más estable y fácil de maniobrar que un hexacopter.

Una vez elegida la configuración hay que hacer una estimación del peso que tiene que levantar el dron. El peso total del dron se estima que será de unos 2 kilogramos. El momento más crítico en que los motores deberán realizar más fuerza es en el ascenso, en este momento para estar seguros de que el dron se va a poder levantar multiplicaremos su peso por 2 y lo dividiremos entre el número de motores para saber la fuerza que debe hacer cada uno de los motores.

$$Fuerza\ motor = \frac{2 \cdot peso\ total}{numero\ motores} = \frac{2 \cdot 2kg}{4} = 1kg \quad (1)$$

Por lo tanto, cada motor debe ser capaz de levantar 1 kilogramo aproximadamente, o lo que es lo mismo 1.000 gramos. Con este dato se pueden empezar a buscar posibles candidatos. A continuación se muestra una tabla con cuatro posibles motores que cumplen el requisito inicial de fuerza que deben levantar. En la tabla aparecen detalladas las características más importantes del motor: el empuje máximo, el amperaje necesario para realizar ese empuje, el voltaje con el que debe ser alimentado y su peso.

Motor	Thrust máx.(g)	Peso(g)	Voltaje(V)	Amperios	Precio(€)
Emax 2216 [25]	960	62	14'8	13	19'90
GT2210 [26]	970	55	11'7	17	12'60
ZTW-4108 [27]	1055	175	14'8	9	32'50
EMX-MT-0070 [28]	950	78	18'8	9	85'00

Tabla 1: Características de los motores candidatos

De los cuatro motores preseleccionados se ha elegido el Emax 2216 ya que sus características hacen que sea el más adecuado para el dron que se desea diseñar. Para la elección del motor se han empleado los métodos PRESS y OWA, para más información sobre el proceso de selección del motor ver anexo C.

La tabla de características del motor elegido es la siguiente:

The voltage	propeller size	current (A)	thrust (G)	power (W)	efficiency (G/W)	speed (RPM)
14.8	APC10*3.8	1	160	14.8	10.8	3420
		2	270	29.6	9.1	4340
		3	360	44.4	8.1	4950
		4	450	59.2	7.6	5440
		5	540	74	7.3	5770
		6	600	88.8	6.8	6100
		7	670	103.6	6.5	6340
		8	730	118.4	6.2	6540
		9	780	133.2	5.9	6810
		10	830	148	5.6	6930
		11	890	162.8	5.5	7130
		12	930	177.6	5.2	7310
		13	960	192.4	5.0	7480
		14	1010	207.2	4.9	7580
		15.4	1070	227.9	4.7	7730
	EMAX1045	1	160	14.8	10.8	3270
		2	260	29.6	8.8	4160
		3	340	44.4	7.7	4830
		4	420	59.2	7.1	5290
		5	500	74	6.8	5680
		6	560	88.8	6.3	6100
		7	630	103.6	6.1	6450
		8	680	118.4	5.7	6700
		9	720	133.2	5.4	6900
		10	770	148	5.2	7100
		11	820	162.8	5.0	7320
		12	860	177.6	4.8	7530
		13	900	192.4	4.7	7710
		14.6	950	216.1	4.4	7890

Figura 10: Características motor Emax MT2216

A partir de los datos de esta tabla se podrán dimensionar el resto de componentes que lleva el dron. Como se muestra en la primera columna el voltaje necesario para alimentar al motor es de 14'8V, esto indica el numero de celdas que debe tener la batería. La segunda columna indica el tamaño de la hélice adecuada para este motor. Se han resaltado dos filas, son las correspondientes a las actuaciones de ascenso y vuelo a punto fijo. Como se ha calculado anteriormente, cada motor debe realizar una fuerza de 1.000 gramos para que el dron pueda ascender, en esta tabla se muestra la correlación entre la fuerza y los amperios de corriente consumidos por el motor. La primera fila resaltada corresponde al vuelo a punto fijo, en la que el dron solamente deberá sostener su propio peso, se han marcado 600 gramos en vez de 500 porque siempre es necesario un margen de seguridad.



Figura 11: Motor Emax MT2216

1.2. Hélice

Para seleccionar la hélice es necesario fijarnos en tabla de características del motor de la figura 10. En la segunda columna nos indica que tipo de hélice necesitamos. La nomenclatura es APC 10x3.8. APC es la marca de la hélice sugerida por el fabricante del motor. El primer número son las pulgadas que debe tener la hélice, en este caso, 10 pulgadas que equivale a 25 centímetros de diámetro aproximadamente. Y el último número indica que paso es aconsejable que tenga la hélice. Por tanto todos los parámetros de la hélice quedan definidos. A continuación se muestra una tabla con las características de la hélice elegida [29].

Diámetro (pulgadas)	10
Paso	3'8
Material	Plástico
Tipo	APC

Tabla 2: Características de la hélice elegida



Figura 12: Hélice APC10x3.8

1.3. Batería

El siguiente componente a seleccionar va ser la batería. La primera columna de la tabla de características del dron (figura 10) nos indica que es necesaria una batería de 14'8 voltios, lo que se traduce en una 4S o de cuatro celdas, ya que cada celda aporta 3'7 voltios y van conectadas en serie. Como se ha explicado en la parte de actuadores, en una batería también es muy importante conocer la capacidad y el ratio de descarga necesarios para realizar el tipo de misión prevista según los motores seleccionados. En el anexo D se calcula cual debe ser como mínimo el valor de estos dos parámetros para que el dron pueda funcionar. La capacidad mínima es de 4A·h y un ratio de descarga de 25C. Se dispone de la siguiente batería:

Capacidad	4.000mAh
Ratio continuo descarga	65C (máx)
Voltaje	14'8V
Peso	535g

Tabla 3: Características de la batería

Sus características se ajustan a los valores calculados. El ratio de descarga es bastante mayor al necesario, puede proporcionar una potencia máxima de 3.848W, cuando se necesitan alrededor de 1.000W para alimentar a los motores, aún y así hay que tener en cuenta que se deberán alimentar otros componentes. Comprando una batería con un ratio de descarga menor podrían ahorrarse unos 100g de peso, aún y así ya que la batería está en perfecto estado se optará por reutilizarla.



Figura 13: Batería BrutePower 4S

1.4. Regulador de velocidad

Para dimensionar el regulador de velocidad, es importante tener en cuenta dos factores: el voltaje de la batería y el amperaje máximo que puede soportar. El voltaje de motores y batería ya ha sido dimensionado previamente, en este caso los motores necesitan 14'8 voltios y por tanto se ha optado por una batería 4S. En cuanto a la corriente máxima que pueden soportar los reguladores hay que trabajar con margen para evitar que en un pico de corriente se quemen estos componentes ya que son muy sensibles. Según los datos de la tabla de características del motor, la máxima corriente con la que pueden ser alimentados es 15'4 amperios. Por lo tanto, buscaremos reguladores que soporten 20-30 amperios. De bajo se muestra una tabla con cuatro posibles reguladores y sus características principales, de entre los cuales se va a escoger el componente utilizado en el dron. Todos los reguladores de la tabla son compatibles con baterías 4S.

Reguladores	Amp. cont.	Amp. máx.	Peso(g)	Precio(€)
RCI Spider 30A OPTO [30]	30	40	24	15'50
Felpa Turnigy 30A [31]	30	40	25	14'04
ZTW Flash 30A BLHeli-S [32]	30	40	7'5	14'99
RCI Spider 30A PRO [33]	30	40	8	15'90

Tabla 4: Características de los reguladores de velocidad preseleccionados

Finalmente, el reguladores elegido ha sido el Flash BLHeli-S. Todos los reguladores propuestos soportan la misma corriente continua y de pico, su principal ventaja respecto a los dos primeros es su peso ligero. El RCI Spider también hubiera sido una buena elección ya que, en este caso, su diferencia de peso es muy pequeña.



Figura 14: Regulador de velocidad ZTW Flash BLHeli-S de 30A

2. Sistemas de control

Una vez elegido el sistema propulsor, el siguiente paso será elegir los componentes que se encargarán de recibir las ordenes e instrucciones sobre las maniobras y rutas que se quieren trazar y ejecutarlas de forma segura y sin comprometer la estabilidad del dron.

2.1. Controladora de vuelo

En el mercado existen diversas posibilidades en cuanto a controladoras de vuelo para drones. Normalmente, las controladoras llevan incorporados y tienen la capacidad de incorporar los sensores necesarios en el vuelo. Como se ha detallado en la parte de sensores, en el caso del dron diseñado son necesarios un acelerómetro, un giroscopio, un magnetómetro, un barómetro y un GPS o la posibilidad de incorporar uno externo. Al elegir la controladora es importante que incorpore o acepte la incorporación de estos componentes. Otra característica importante es la compatibilidad con el tipo de aeronave, es decir, si es solo para quadcopters, se puede usar en cualquier tipo de multicopter o incluso en aviones. Las principales opciones son: Pixhawk 4, Naza-M v2, ArduPilot Mega. A continuación se detallarán las ventajas e inconvenientes de cada una de ellas para poder tomar una decisión. También se han tenido en cuenta modelos como la Omnibus F4 o CC3D pero se han descartado puesto que están diseñadas para controlar racers y por tanto no se adaptan a los requisitos de este proyecto en concreto.

Naza-M v2

Es una controladora de la empresa DJI, empresa especializada en drones. Es compatible solo para este tipo de vehículos en las configuraciones de quadcopter, hexacopter y octocopter. Una de sus ventajas es su facilidad de configuración y el poder garantizar un vuelo muy estable y a la vez una gran maniobrabilidad [34]. Estas son sus principales características:

- Sensores de abordo: giroscopio de 3 ejes, acelerómetro de 3 ejes y barómetro
- GPS externo opcional

- Compatible con receptores de salida D-Bus/S-Bus y PPM compuesta
- Incluye funciones de control y orientación inteligente (Intelligent Orientation Control) y la función de regreso al hogar (Return-To-Home)
- Peso: 27g
- Dimensiones: 45.5 x 32.5 x 18.5mm

ArduPilot Mega

Esta controladora está basada en la plataforma Arduino Mega. Está pensada para incorporarse en vehículos autónomos ya que es capaz de estabilizarse autónomamente y tiene un sistema de navegación basado en way-points. Es compatible con alas volantes, multicopters, helicópteros e incluso rovers. Posee un sistema de código abierto y, aunque no es necesario, se puede programar con Arduino [35]. Sus principales características son:

- Los chips ATMEGA2560 y ATMEGA32U-2 de Atmel para procesamiento y funciones USB respectivamente
- Sensores de abordó: giroscopio, acelerómetro (DoT MPU-6000 de Invensense) y magnetómetro de 3 ejes, junto con un barómetro de alto rendimiento (MS5611-01BA03)
- GPS externo opcional, módulo uBlox LEA-6H con brújula
- Peso: 31g
- Dimensiones: 70.5 x 45 x 13.5mm

Pixhawk 4

Pixhawk es un proyecto independiente de hardware abierto que intenta proporcionar una controladora de vuelo de alta calidad y a bajo coste, tal y como se especifica en su página web. Puede utilizarse para controlar cualquier multicopter, avión, rover o incluso barco [36]. Sus principales características son:

- Procesador principal de FMU: STM32F765 de 32 Bit Arm Cortex-M7, velocidad 216MHz, 2MB de memoria, 512KB de RAM
- Procesador IO: STM32F100 32 Bit Arm Cortex-M3, velocidad 24MHz, 8KB de memoria SRAM
- Sensores de a bordo: Acelerómetro/giroscopio ICM-20689, acelerómetro/giroscopio BMI055, magnetómetro IST8310 y barómetro MS5611
- GPS: ublox receptor Neo-M8N GPS / GLONASS; magnetómetro integrado IST8310
- Peso: 15.8g
- Dimensiones: 44 x 84 x 12mm

Finalmente se ha optado por esta última opción, la Pixhawk 4. Es una controladora muy versátil que se puede usar en todo tipo de vehículos, terrestres, aéreos o acuáticos. Además, el paquete incorpora un GPS y una placa distribuidora. La Ardupilot Mega ha sido descartada puesto que fue la antecesora de la Pixhawk y actualmente no se reciben actualizaciones. La Naza-M v2 también ha sido descartada por su dificultad de programación.



Figura 15: Controladora Pixhawk 4

2.2. Ordenador de abordo

Una vez elegida la controladora, es necesario un ordenador de abordo que pueda transmitirle información para guiar al dron. Los requisitos más importantes de este componente son que ha de ser capaz de comunicarse con la controladora, fácilmente programable y de peso y tamaño reducidos ya que en el caso de los drones el peso siempre es un factor clave.

El ordenador de abordo seleccionado para este trabajo es la Raspberry Pi 3 Modelo B. Es un ordenador fácilmente programable pensado para la enseñanza en las escuelas. Su sistema operativo oficial es Raspian, diseñado específicamente para Raspberry Pi y en el que sus ingenieros trabajan para optimizarlo. Aún así, se posible substituirlo por otros siendo Linux el más recomendado en caso de querer realizar este cambio [37]. Sus principales características son las siguientes:

- Procesador Quad Core Broadcom 2837 ARMv8 64bits
- Velocidad del procesador: 1,2 GHz
- 1 GB RAM
- BCM43143 WiFi integrada
- Bluetooth Low Energy (BLE) integrado
- 40 pines de expansión GPIO
- 4 puertos USB2
- Salida de vídeo y estéreo
- Puerto cámara CSI para conectar la cámara Raspberry Pi
- Puerto de visualización DSI para conectar la pantalla táctil Raspberry Pi
- Lector de tarjeta Micro SD para cargar un sistema operativo y almacenar sus datos
- Fuente de alimentación Micro USB (soporta hasta 2,5A)



Figura 16: Raspberry Pi 3 Modelo B

3. Sistemas de comunicación

Es necesario poder establecer un modo de comunicación entre la estación de control en tierra y el dron tanto para poder mandarle ordenes como para recibir información sobre su estabilidad, posición y velocidad entre otros. Por este motivo se necesitan incorporar una serie de componentes que permitan este tipo de comunicación.

3.1. Antena

Las lecturas de los sensores nos permiten conocer el comportamiento del dron. Para poder tener acceso a estas lecturas en tiempo real y así poder tomar decisiones sobre las actuaciones del dron se requiere un sistema de comunicación entre la estación de control en tierra y el dron. De esta forma, son necesarios dos módulos de radio telemetría, uno irá incorporado en el dron y conectado a la controladora y el otro se conectará a la estación en tierra. De esta forma es posible la comunicación entre ambos módulos.



Figura 17: Módulos de radio telemetría

3.2. Receptor

Se ha optado por la utilización de un sistema de control desde tierra en caso que el control autónomo falle y se necesite corregir o tomar el control de la aeronave. Para ello será necesario un receptor y un mando a distancia o transmisor.

Tanto el receptor como el mando se utilizarán del material disponible en el proyecto Aldora. En el caso del receptor es importante tener en cuenta dos aspectos: el numero de canales del que dispone y su compatibilidad con el mando de radio control. En este caso es suficiente con seis canales. Cuatro de ellos para los movimiento de guiñada, alabeo y cabeceo y para la aceleración. Los otros dos se reservarán para sistemas de seguridad. El receptor elegido es el FS-IA6B, sus principales características son las siguientes:

Canales	6
Rango de frecuencia	2.4 - 2.48 GHz
Voltaje de entrada	4.0 - 6.5V DC
Peso	14.9 g

Tabla 5: Características receptor FS-IA6B



Figura 18: Receptor FS-IA6B

En cuanto al mando o unidad transmisora, el aspecto más importante es que sea compatible con el receptor seleccionado con tal de que puedan ser emparejados. En este caso se utilizará el modelo FS-i6X que es de la misma marca del receptor, asegurando así su compatibilidad.



Figura 19: Transmisor FS-i6X

4. Sensores

Además de los sensores que la controladora Pixhawk 4 lleva incorporados, será necesario añadir alguno para mejorar sus prestaciones y para la adquisición de datos necesarios para el correcto comportamiento del dron.

4.1. GPS

Como se ha dicho anteriormente, el GPS nos proporciona con gran exactitud la posición del dron en cada momento, además es capaz de controlar la altitud del dron, al igual que el barómetro pero con mayor exactitud, eso hace que sea un sensor indispensable. Nos ayudará también a conocer su velocidad aunque no sea un dato clave en esta aplicación. No será necesario realizar un proceso de selección de este sensor ya que la controladora seleccionada incluye un kit que contiene además de la propia controladora, un GPS, un soporte para situar el GPS en alto y así evitar interferencias con otros componentes y, por último, una placa distribuidora en la que conectar los reguladores de velocidad y doble salida de BEC de 5 voltios.



Figura 20: Kit Piwhawk 4

5. Componentes adicionales

En el apartado de análisis de la normativa se especifican una serie de componentes que el dron debe incorporar. En este apartado se detallarán los componentes que no han sido descritos anteriormente y que son necesarios para cumplir con la normativa vigente en el estado español.

5.1. Dispositivo de visibilidad

Según se especifica en la normativa es necesaria la incorporación de algún tipo de dispositivo o pintura que asegure la visibilidad de la aeronave. En este caso se optará por la colocación de luces led que se encenderán al conectar el dron, de esta forma podrá ser visto por otros pilotos en todas las fases del vuelo, incluso en tierra. Se han seleccionado tiras de luces RGB programables para poder señalar además la posición del dron, es decir, los dos brazos delanteros estarán iluminados por luces de un color y los brazos traseros por otro. La placa de led elegida para esta aplicación es la Matek 2812ARM-4 [38], cuyas características son:

- Cada placa incluye 4 led
- Posibilidad de conexión en serie (DIN - DO)
- Completamente programable (tanto colores como modos de funcionamiento)
- Se pueden configurar hasta 7 colores: blanco (9W); verde, rojo y azul (3W); amarillo, cian y magenta (6W)
- Alimentación a 5V
- Peso: 6g



Figura 21: Placa 4 LED RGB Matek Systems

5.2. Paracaídas

Otro requisito impuesto por la normativa si el dron va a sobrevolar aglomeraciones de edificios en ciudades, pueblos o lugares habitados o reuniones de personas al aire libre necesitará un dispositivo que limite su energía de impacto. En este proyecto, las pruebas se realizarán en un campo de vuelo por lo que no sería obligatoria la incorporación de este dispositivo. Aún así, se ha diseñado un sistema de paracaídas que podría incorporarse al dron en un futuro.

El sistema consta de dos paracaídas, uno de tamaño pequeño cuya apertura provocará o ayudará a abrir otro paracaídas de mayor tamaño, este segundo paracaídas es capaz de frenar la caída del dron. La apertura del paracaídas se realizará de forma manual a través del control remoto en caso de que se produzca algún tipo de error. Esta se hace a través de un servoactuador conectado a uno de los canales de la unidad de control remoto. El peso de todo el conjunto del sistema de paracaídas es de 425 gramos.

5.3. Cámara

En la normativa también se especifica que si el dron vuela fuera del alcance visual del piloto es necesario incorporar una cámara o dispositivo de visión orientado hacia delante. En este caso será necesario, además de la cámara, el diseño del soporte que ayude a orientarla hacia delante para poder observar hacia donde esta yendo el dron. Puesto que el ordenador de abordo seleccionado es la Raspberry Pi se propone utilizar una cámara para Raspberry, en concreto la Raspberry Pi Camera Module v2 [39], debido a su reducido tamaño y peso. Sus principales características son las siguientes:

- Resolución nativa de 8MP
- Resolución fotos de 3280x2464
- Resolución vídeo 1080p30, 720p60 640x480p90
- Tamaño: 25mm x 23mm x 9mm
- Peso: 3g

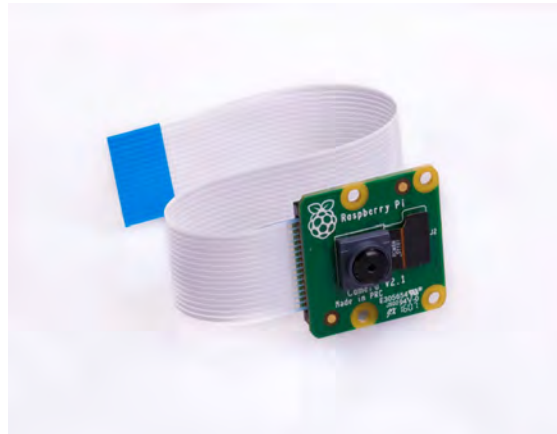


Figura 22: Raspberry Pi Módulo Cámara v2

Parte VII

Prototipo

En este apartado se detallarán las características del prototipo que incorporará los componentes seleccionados en el apartado anterior. Se han añadido los componentes seleccionados a un dron existente que pertenece al proyecto Aldora. También se mostrará un esquema de conexiones de todas sus partes así como donde estarán situadas en el prototipo.

1. Características

Las características más importantes de un dron son principalmente su peso y su configuración. En cuanto a la configuración, al dimensionar se optó por incorporar 4 motores por lo que el prototipo será un quadcopter. Otro aspecto a tener en cuenta va a ser la situación de estos motores, es decir, que estructura van formar los brazos que sujetarán a los motores. Las configuraciones más habituales que puede tener un quadcopter son las siguientes: en X, en +, en V o en H. También pueden plantearse otras configuraciones como la V tail y la A tail, aunque en estas los motores tienen un cierto ángulo de inclinación y no se utilizan tanto [40].

El caso del prototipo utilizado, tiene una configuración en X, tal y como se muestra en la figura 23, este aspecto es muy importante al realizar las conexiones puesto que la numeración y conexión de los motores depende de la configuración. Otro aspecto muy importante va a ser su orientación, tanto la controladora de vuelo como el GPS deben ir orientados hacia delante, es por eso que es necesario distinguir bien los brazos delanteros de los traseros incluso en vuelo para poder controlar bien el dron. En el caso de este prototipo en concreto, los brazos delanteros son blancos, mientras que los traseros son rojo y negro; además, las placas de luces led incorporadas para asegurar la visibilidad son de distinto color en los brazos delanteros y traseros asegurando así poder distinguir siempre su orientación en el aire.

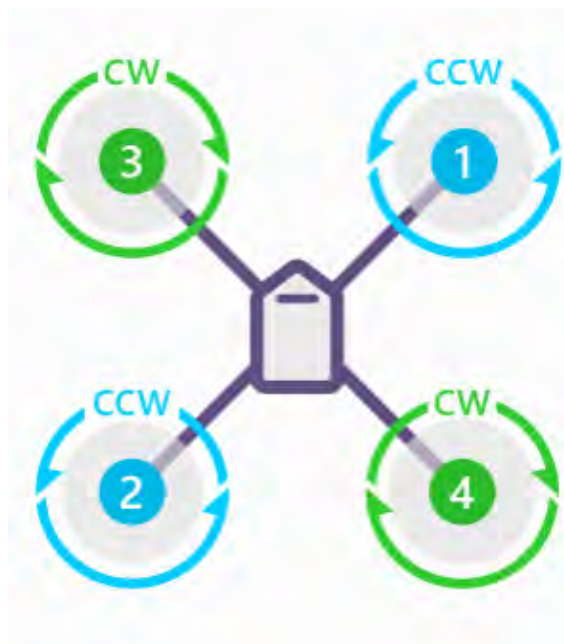


Figura 23: Configuración del prototipo

En cuanto a su peso, es de 1'640 kilogramos sin incorporar el sistema de paracaídas. Se había dimensionado para un peso de 2 kilogramos, incorporando el paracaídas se alcanza ese peso, validando así los cálculos realizados en el apartado anterior. A continuación se muestran imágenes del dron en las que se puede observar la distribución de los componentes en el chasis.

En la primera imagen se puede ver uno de los brazos del dron. En él se colocan un motor y el regulador de velocidad al que va conectado. En el extremo del brazo se puede observar una una pieza naranja que une el brazo del dron con una barra de carbono, esta barra de carbono sirve como tren de aterrizaje. Se ha optado por utilizar barras de carbono ya que resisten mejor los impactos contra el suelo en comparación con las posibles alternativas impresas en 3D.



Figura 24: Brazo del dron

En la siguiente imagen se pueden observar los cuatro brazos unidos a la parte central del dron. Excepto los cuatro brazos y las barras de carbono usadas como tren de aterrizaje, el resto de piezas utilizadas han sido impresas en 3D. Como se ha comentado anteriormente, se utiliza una configuración en X. Para poder distinguir su disposición en el aire los brazos son de distinto color, los delanteros son blancos y los traseros rojo y negro. En la protección de los reguladores de velocidad se encuentra la numeración que aparece en la figura 23 para facilitar el proceso de conexión.



Figura 25: Montaje de los cuatro brazos

En un lateral del dron se coloca una pieza que sirve de soporte para el GPS, la antena de comunicación y el receptor. El GPS se coloca en el soporte que incluye el kit de Pixhawk 4 elegido, que aparece en la figura 20, este soporte se atornillará a la pieza impresa en 3D. Esta pieza consta de dos cavidades, en una de ellas se introduce el módulo de telemetría que debe ir en el vehículo. Por el otro extremo, no visible en la imagen, hay un orificio por el que sale la antena. Para evitar posibles movimientos del módulo de comunicación dentro de la cavidad se utilizará espuma para que quede bien fijada. Por último, en la cavidad inferior se encuentra la unidad receptora. En este caso no hace falta utilizar ningún complemento para fijarla puesto que gracias a su forma al insertarla en la cavidad queda completamente sujeta.

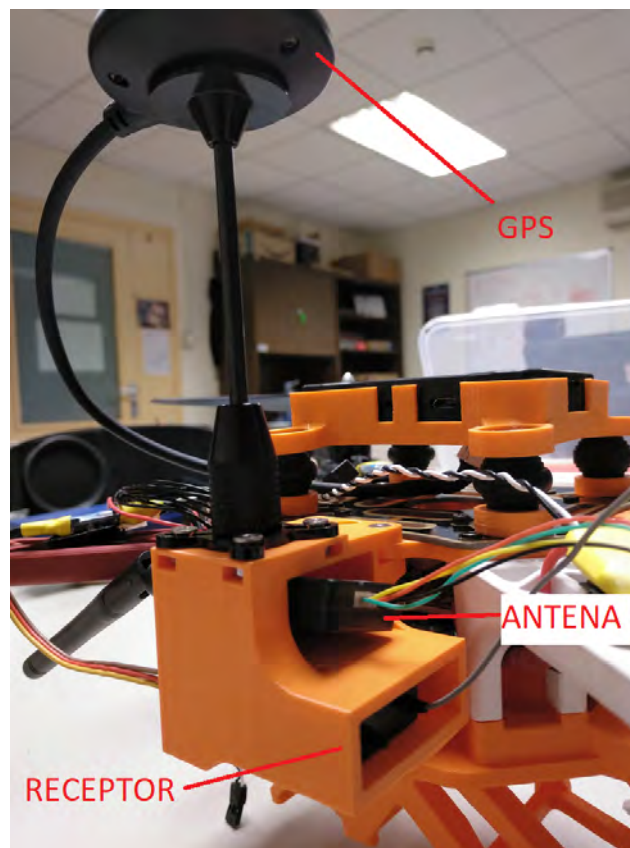


Figura 26: Situación del GPS, antena y receptor

Los componentes situados en la parte central del dron son: la controladora de vuelo Pixhawk 4, la placa distribuidora y el ordenador Raspberry Pi 3. La controladora de vuelo se sitúa en la parte superior, esta fijada con cinta de doble cara a una carcasa, impresa también en 3D, que se adapta perfectamente a su silueta. Esta carcasa se une al dron por medio de unas gomas antivibración. El motivo por el cual se usa este método es para evitar posibles errores, es decir, al recibir los datos de los sensores puede cometer errores al procesarlos a causa de su constante estado de vibración y enviar unas ordenes erróneas a los actuadores que desestabilicen el dron o no cumplan con la función deseada. La solución para minimizar el riesgo de errores es evitar esas vibraciones por medio de gomas antivibración. La placa distribuidora está situada en el nivel intermedio, también dispone de una carcasa que se adapta a su forma a la cual va unida por medio de cinta de doble cara. En ella se conectarán los reguladores de velocidad y también subministrará energía a la Pixhawk y la Raspberry. Por tanto, su situación en el dron debe permitir que las conexiones se realicen con facilidad. Finalmente, la Raspberry Pi ocupa el nivel inferior. Del mismo modo que las otras placas también dispone de una carcasa a la cual va unida. Su carcasa se puede extraer por la parte inferior del dron, cuando no esta puesta la batería, encaja con el agujero rectangular de la figura 25 y va atornillada al soporte que aparece en esa imagen.

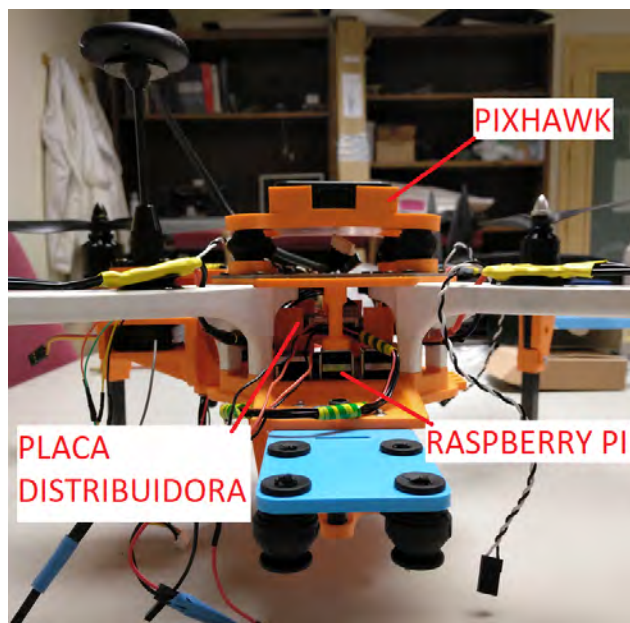


Figura 27: Situación de la controladora, placa distribuidora y ordenador de a bordo

La batería, puesto que es el último elemento a colocar, se encuentra en la parte inferior, por debajo del resto de componentes. Dispone de su propio elemento de sujeción que diseñado acorde con sus dimensiones, no es una pieza maciza porque, en consecuencia con su gran tamaño, conllevaría en un aumento del peso innecesario. Por tanto, la pieza puede soportar el peso de la batería y mantenerla fija en esa posición durante el vuelo, siendo fácil de introducir y extraer este componente. Para evitar que salga por los extremos, por los cuales se introduce, se usan dos barras de carbono que realizarán la función de tope y no permitirán que se mueva dentro de su soporte.

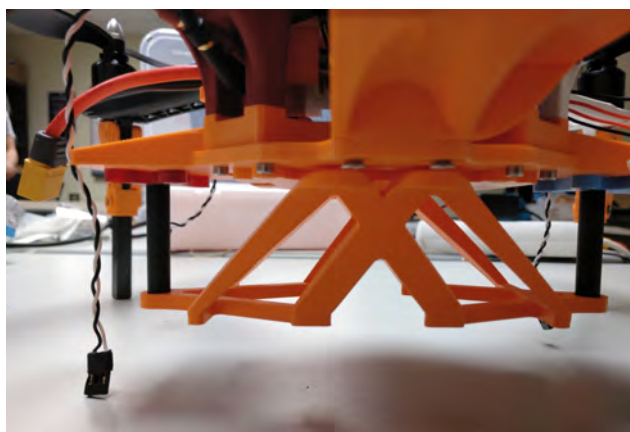


Figura 28: Situación de la batería

Las luces, utilizadas como dispositivos de visibilidad, están formadas por cuatro placas de cuatro led cada una de ellas. Cada una de las placas se sitúa en la parte inferior de los cuatro brazos que forman el dron, por medio de cinta de doble cara. Se han incorporado al dron una vez finalizado el montaje del resto de partes ya que es necesario aumentar la longitud de los cables. Las placas deben ir conectadas en serie y eso supone pasar los cables por alrededor de toda su estructura, así como asegurar (por medio de bridas) que los cables queden bien sujetos y que no haya riesgo que ninguno de ellos pueda enredarse con las hélices durante el vuelo. Como se ha comentado, los led son programables, de esta manera las dos placas situadas en los brazos delanteros, los de color blanco, emitirán luz de color verde; mientras que las dos placas colocadas en los brazos traseros emitirán luz roja.



Figura 29: Situación de las luces led

El soporte de la cámara esta compuesto por dos piezas. La primera de ellas es la que se une a la cámara Raspberry Pi, dispone de un agujero rectangular para la lente y cuatro redondos para poder atornillar la cámara al soporte. La forma de esta pieza permite orientar el dispositivo hacia delante, cumpliendo así con la normativa vigente en caso de volar fuera del alcance visual del piloto. La segunda pieza va unida al dron, también por medio de tornillos, debajo del cuerpo del dron y atravesada por la barra de carbono que impide el movimiento de la batería. Las dos piezas se unen con gomas antivibración, estas gomas ayudarán a reducir las vibraciones y permitirán una mejor calidad de vídeo y por tanto una mejor visión de hacia donde se dirige el dron. Para poder pasar el cable y conectarlo en la ranura correspondiente de la Raspberry, ambas piezas disponen de una pequeña ranura por donde pasar el cable y facilitar así esta conexión.



Figura 30: Soporte de la cámara Raspberry

Esta es la imagen final del dron una vez montado y con todos sus componentes fijados y colocados en el lugar que le corresponde dentro del chasis. Todas las conexiones se han realizado, tal y como se explicará en el siguiente apartado, y los cables demasiado largos han sido unidos a la estructura por medio de cinta adhesiva o bridas para evitar que puedan alcanzar alguna de las hélices durante el vuelo y minimizar así el riesgo de accidentes.



Figura 31: Montaje final

2. Conexiones

Una vez conocidas las características del prototipo, el siguiente paso será la conexión de todos sus componentes. Para empezar, el dron dispone de 4 motores y 4 reguladores de velocidad, cada motor deberá conectarse a un regulador. Estos reguladores se conectarán a placa distribuidora, el positivo y negativo irán conectados en las esquinas de la placa, usando los conectores denominados bullets; mientras que el cable de señal debe conectarse en uno de los pines 1-4 según la numeración que se muestra en el apartado anterior referente a una configuración en X, figura 23. Como se puede observar en el esquema, la placa distribuidora también dispone de un conector XT60 en el cual se conectará la batería, es importante que esta conexión se realice cuando todo esté listo para volar, con las hélices bien sujetas y los motores desarmados.

La placa distribuidora se conectará a la controladora por medio de dos cables, el primero es el power1, a través del cual se dará alimentación a la controladora de forma que esta será capaz de alimentar a su vez los sensores que se conecten a ella. Además, se conectará otro cable con entrada/salida PWM desde el cual se controlará la intensidad que es necesario proporcionar en cada momento a cada uno de los motores conectados a la placa distribuidora. En el esquema se puede apreciar una tercera conexión, desde el power2, con la Raspberry Pi y las placas de luces led que se explicará más adelante.

Posteriormente, con la controladora alimentada, se realizará la conexión de los sensores y actuadores restantes. El GPS se conectará a la entrada de GPS module, es muy importante que el GPS se oriente en la misma dirección que la controladora, en ambos componentes hay dibujada una flecha que indica la orientación en la que deben ser colocados. En cuanto a la telemetría, la antena emisora de datos se conecta a la controladora en la entrada telem1, mientras que la antena receptora irá conectada a la estación de tierra, en este caso un ordenador. El siguiente componente a conectar es el receptor, elemento que permitirá tomar el control del vehículo y que este sea controlado a través de un transmisor (mando). El receptor se conectará a la entrada PPM/RC de la controladora de vuelo.

El último paso es conectar el ordenador de a bordo y los componentes conectados a él. Para ello es necesario realizar dos conexiones, una desde la controladora y la otras desde la placa distribuidora.

La primera conexión con la controladora se realiza al telem2, desde el cual saldrán tres cables: negativo, RX y TX que en el ordenador (Raspberry) se conectarán en los pines correspondientes al ground, GPIO 15 Y GPIO 14, respectivamente, de esta forma el ordenador podrá comunicarse con la controladora. La segunda conexión, desde la placa distribuidora, es la que dará alimentación a la Raspberry, no se realiza por medio del telem2 de la controladora porque la intensidad suministrada no es suficiente para que la Raspberry realice sus funciones correctamente. Por tanto, esta conexión se realizará desde el power2 de la placa distribuidora, de allí saldrán dos cables positivos y dos negativos; un positivo y un negativo se conectarán a los pines de 5V y ground de la Raspberry. En cuanto a los cables positivo y negativo restantes del power2 se utilizarán para alimentar las luces led que disponen de estos dos cables y un tercero, de señal, que irá conectado al pin GPIO18 de la Raspberry. Finalmente, solo queda realizar la conexión de la cámara, que se conectará a la entrada camera de la Raspberry.

Toda la información referente a las conexiones de la controladora Pixhawk 4 y el ordenador de abordo Raspberry Pi modelo B se pueden encontrar tanto en los documentos adjuntos en el anexo de fichas técnicas (anexo F) como en las siguientes direcciones web [36], [41] y [42]. Para facilitar la comprensión de este apartado se adjunta un esquema en el que aparecen todos los componentes que forman parte del prototipo y la forma en la que están conectados.

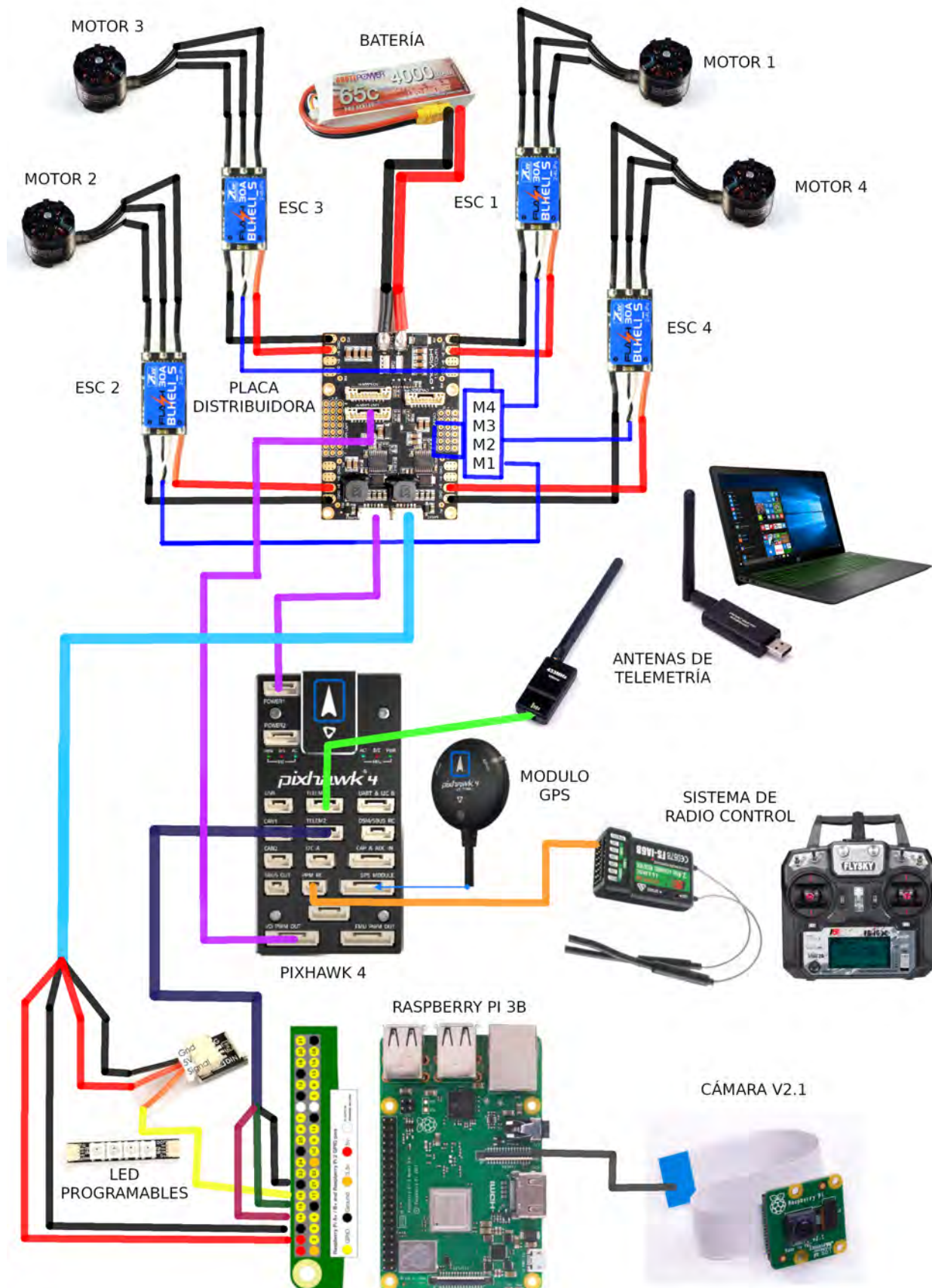


Figura 32: Esquema de conexiones

Parte VIII

Entorno de programación

Una vez seleccionados los componentes que formarán parte del dron, nos centraremos en la controladora y el ordenador de abordaje elegidos así como en sus comunicaciones con la estación de control en tierra.

1. Software

La controladora de vuelo Pixhawk 4 utiliza un software llamado Ardupilot. Ardupilot es un software de código abierto de autopiloto, es el único capaz de controlar cualquier tipo de vehículo, ya sea aéreo (avión, ala volante, multirrotor, etc.), terrestre (rover) o acuático (barco o submarino) [43]. Además, este software es el más utilizado y por tanto también el más testado. Al tener una base de código abierto puede evolucionar al mismo ritmo que la tecnología. Por lo que respecta a su uso en drones, lo utilizan empresas como 3DR, jDrones o PrecisionHawk. También ha sido utilizado por grandes instituciones como la NASA, Intel o Insitu/Boeing para hacer pruebas y desarrollar tecnología.

2. Sistema de comunicación

Como se ha explicado anteriormente, se producirán dos tipos de comunicaciones. Por un lado, es necesaria la comunicación entre el ordenador de abordaje, que es el que contiene las ordenes previamente programadas, y la controladora de vuelo, que es la encargada de que se ejecuten y que el vuelo sea estable. Por otro lado, también existirá una comunicación entre la estación de control en tierra y el ordenador de abordaje en el que se podrán introducir cambios en la ruta y se recibirán datos sobre el estado del dron.

Todas estas comunicaciones se producirán a través de un protocolo de comunicación conocido como MAVLink (Micro Air Vehicle Link). MAVLink es un sistema de comunicación utilizado para establecer comunicación con los drones desde la estación de control en tierra y también entre los componentes que constituyen el dron [44]. Es compatible con muchos lenguajes de programación como por ejemplo C, C++, Python, Java o JavaScript entre otros soportados en diferentes sistemas operativos como Windows, Linux, MacOS, Android o iOS.

3. Estación de control en tierra

Es necesaria una estación de control en tierra desde donde poder planificar una ruta y tener un seguimiento del comportamiento del dron a tiempo real. Para ello se utilizará QGroundControl. *QGroundControl es una plataforma que proporciona un control completo del vuelo y permite planificar misiones para cualquier tipo de dron con MAVLink* [45]. Es compatible con los sistemas operativos Windows, OS X, Linux, iOS y Android. Soporta diferentes tipos de autopilotos como PX4 Pro, Ardupilot o cualquier otro tipo de vehículo que utilice el protocolo de comunicación MAVLink. Funciona con todos los tipos de vehículos compatibles con Ardupilot y PX4 Pro, es decir, los ya mencionados anteriormente: multirrotores, aviones, helicópteros, alas volantes, rovers, barcos y submarinos.

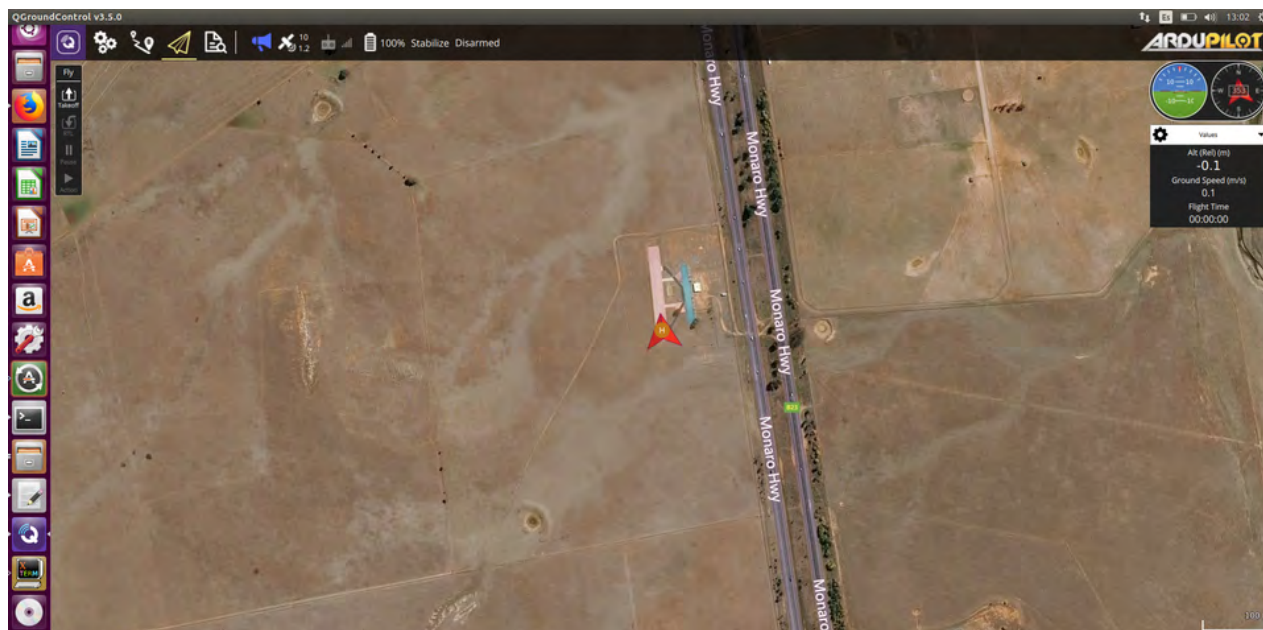


Figura 33: Estación de control QGroundControl

4. Lenguaje de programación

Para la realización de este trabajo se utilizará Python como lenguaje de programación. Es un lenguaje orientado a objetos con el que se pueden realizar muchos tipos de programas incluso desarrollo de páginas web. Se trata de un lenguaje interpretado, es decir, que no es necesario compilar el código fuente para ejecutarlo. Sus principales ventajas son la gran cantidad de librerías que contiene, los datos y funciones que incorpora, que ayudan a realizar programas de forma más rápida sin necesidad de programarlo todo desde cero. Además, es gratuito y compatible con muchas plataformas.



Figura 34: Logotipo Python

La elección de Python como lenguaje de programación para desarrollar este trabajo, además de su sencillez con respecto a otros lenguajes potenciales como C++, es poder utilizar las librerías de DroneKit. DroneKit es una API (Application Programming Interface) en Python, creada por 3D Robotics, cuyo objetivo es el desarrollo de aplicaciones que se ejecutan desde el ordenador de a bordo y la comunicación con la controladora de vuelo que incorpora el software Ardupilot [46]. La API se conecta con los vehículos a través del protocolo de comunicación MAVLink, descrito anteriormente. Gracias a esta comunicación se recibe información referente a la telemetría, el estado del vehículo y sus parámetros. Estos datos permiten gestionar una misión y controlar el movimiento y las operaciones del dron. DroneKit es compatible con los sistemas operativos Windows, Linux y Mac OS. Las aplicaciones desarrolladas con DroneKit en Python suelen ejecutarse en un ordenador de a bordo en Linux.

Parte IX

Pruebas en tierra

En este apartado se detallarán todas las pruebas y programas realizados para conseguir completar un programa final funcional que cumpla con los requisitos impuestos: que el vehículo sea capaz de despegar, volar hasta unas coordenadas determinadas, aterrizar en ellas para realizar la entrega de un paquete y volver al lugar de origen. Como se detallará a continuación, para obtener un programa final funcional es necesario realizar una serie de simulaciones antes de probar el programa con el prototipo para asegurar que funciona correctamente. También se intentará simplificar realizando pequeños programas de prueba que, una vez validados por separado, podrán juntarse y formar el programa definitivo. Seguidamente se detallarán los pasos seguidos en el ámbito de la programación.

1. Simulaciones

Antes de realizar pruebas con el prototipo, es necesario realizar una serie de simulaciones para comprobar que los programas funcionan correctamente. De esta forma se evitarán posibles fallos debidos a errores de programación, es decir, es posible que el programa no contenga errores de código y se pueda ejecutar correctamente pero quizás no realiza las funciones de forma adecuada y esto es lo que se intenta evitar mediante las simulaciones.

Para poder realizar simulaciones es necesario un software que simule las prestaciones de un quadcopter, el software seleccionado es SITL (Software in the loop). SITL es un código de autopiloto que usa un compilador de C++, crea un ejecutable que permite probar el comportamiento de un código sin el hardware [47]. Este software, permite ejecutar Ardupilot directamente en un ordenador sin necesidad de utilizar un hardware especial, puesto que Ardupilot es autopiloto que se puede ejecutar en una gran variedad de plataformas. De esta forma el ordenador se convierte en una plataforma más en la que Ardupilot puede compilarse y ejecutarse.

Al ejecutar SITL los datos de los sensores provienen de un modelo dinámico de vuelo de un simulador de vuelo. Ardupilot lleva integrados una gran variedad de simuladores de vehículos y, además, puede interactuar con simuladores externos. Debido a esta versatilidad, Ardupilot puede probarse en diferentes tipos de vehículos. En este caso, SITL puede simular: aviones multi-rotor, aviones de ala fija, vehículos terrestres, vehículos submarinos, gimbals (cámara), rastreadores de antenas y una gran variedad de sensores opcionales como Lidars y sensores de flujo óptico. El entorno Ardupilot SITL se ha desarrollado para ejecutarse de forma nativa tanto en Linux como en Windows. A continuación se muestra una imagen de la arquitectura del simulador (los números de los puertos que aparecen son orientativos).

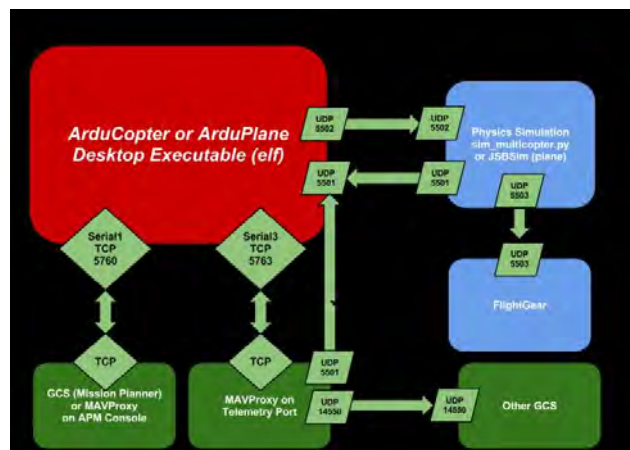


Figura 35: Arquitectura de Ardupilot en SITL

A continuación se explicarán de forma detallada las simulaciones realizadas:

1.1. Misión con QGroundControl

La primera simulación realizada ha sido con la estación de control QGroundControl. Para ello es necesario abrir la imagen del programa y, una vez abierto, ejecutar el simulador tal y como aparece explicado en el siguiente enlace [48]. De esta forma la estación de control detecta que se ha conectado un vehículo y se pueden monitorizar todas sus prestaciones y realizar misiones. En la figura 36 en la que aparece el programa abierto y los comandos utilizados desde la terminal de Linux para ejecutar el simulador. Para ejecutar el simulador hay que abrir la carpeta donde se encuentra instalado y dentro de ella se encontrará la opción de simular una multi-rotor o un avión, en nuestro caso simularemos un multi-rotor, y desde allí se ejecutará el programa de simulación con el comando `sim_vehicle.py`.

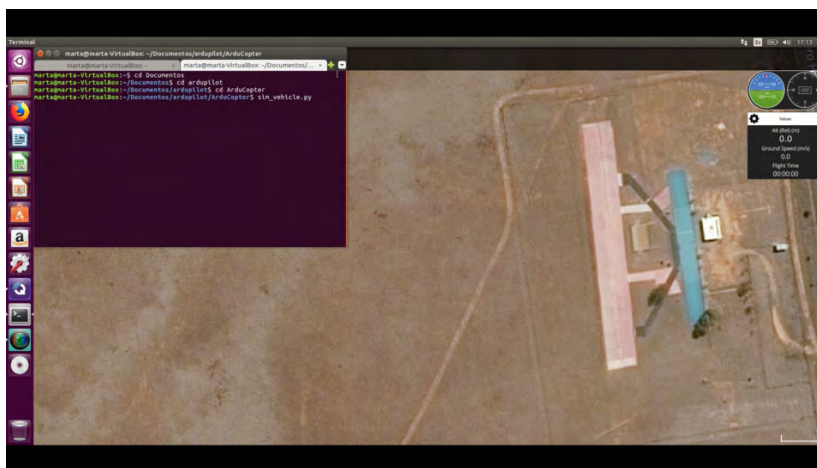


Figura 36: Simulación de quadcopter desde QGroundControl

Cuando la estación de control detecte que el vehículo ha sido conectado se podrá proceder a planear una misión. La misión que se simulará se dividirá en 5 pasos. El primer paso es el despegue hasta una altura determinada (por ejemplo 10 metros). Cuando el vehículo se encuentre a 10 metros del suelo deberá volar hasta un punto seleccionado en el mapa y aterrizar en él. El tercer paso consiste en detener la misión 30 segundos una vez el vehículo haya aterrizado en el punto de destino, tiempo suficiente para entregar el paquete que transporte. El cuarto paso consiste en volver a despegar hasta la altura seleccionada (10 metros nuevamente). Y, por último, el vehículo regresará al lugar desde el que ha despegado la primera vez y aterrizará allí de nuevo. En la siguiente imagen se puede ver un esquema de la misión descrita.

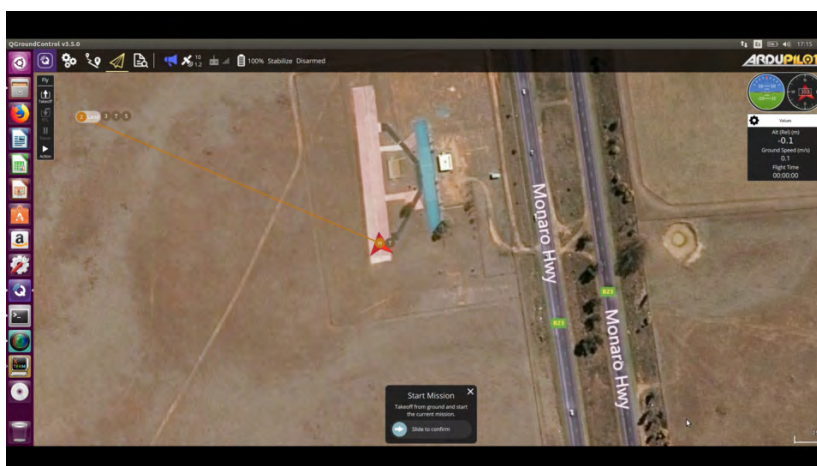


Figura 37: Esquema de misión en QGroundControl

Para realizar la simulación de esta misión simplemente es necesario deslizar la flecha que aparece en la parte inferior de la pantalla. Una vez hecho el vehículo se armará y empezará la misión. Por pantalla se mostraran datos referentes al estado del dron. En el margen izquierdo se encuentra un cuadro con los datos de altura a la que se encuentra el vehículo del suelo en metros, velocidad respecto a tierra en metros por segundo, y tiempo de vuelo en horas : minutos : segundos; así como una brújula que muestra la orientación del vehículo según los puntos cardinales y un horizonte artificial que muestra la actitud. En la parte superior de la pantalla se muestra el porcentaje de batería restante, el modo de vuelo (en este caso auto) y si el dron se encuentra armado o desarmado.

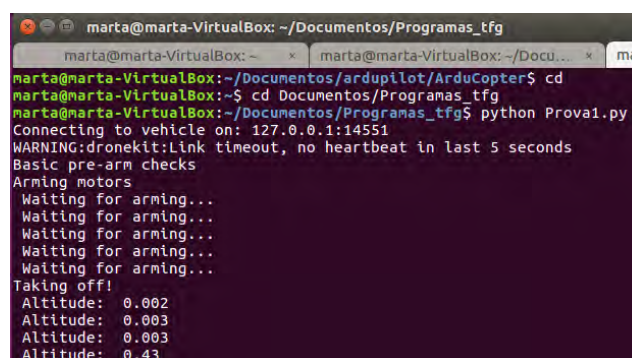
Después de haber realizado con éxito la esta simulación, el siguiente paso es la programación de scripts en Python. El objetivo es poder realizar la misma misión que hemos simulado con QGroundControl pero esta vez con el código escrito desde Python. Para facilitar este proceso se harán diversos programas, cada programa realizará una o varias de las funciones que debería realizar el código completo. De esta forma se facilita la programación ya que es más fácil comprender las funciones por separado y además facilita la detección de errores.

Como se ha comentado en el apartado anterior, Python dispone de la librería DroneKit. Esta librería incluye programas de ejemplo y funciones ya definidas como por ejemplo el despegue, estos ejemplos y funciones han sido estudiados para el desarrollo de los programas que se encuentran en este trabajo. A continuación, se detallarán los programas realizados con Python y sus respectivas simulaciones:

1.2. Despegue y aterrizaje reiterado

Este primer programa consiste en simplemente hacer despegar al dron, a una altura aproximada de 10 metros, una vez lograda esa altitud, aterrizará en ese mismo punto desde el cual despegó. Cuando se encuentre en el suelo se detendrá durante 20 segundos, tiempo suficiente para poder hacer la entrega de un paquete. Después de la pausa volverá a despegar y cuando se encuentre a 10 metros del suelo aterrizará de nuevo. Este programa sirve para comprender las funciones de despegue y aterrizaje y familiarizarse con los ejemplos de la librería DroneKit. El código utilizado se encuentra en el anexo E.1.

Para simular este código es necesario ejecutar el simulador, de la misma forma que en la anterior simulación, desde la terminal de Linux se abre la carpeta donde se encuentra ArduPilot y se ejecuta el simulador de un multi-rotor. Una vez se ha iniciado el simulador y están listas todas sus funciones se deberá abrir otra terminal, en ella buscar el programa a simular y ejecutarlo con el comando python espacio_nombre_del_programa.py. Al pulsar enter el programa empezará a simularse y se mostrarán por la terminal de Linux los mensajes se han designado para mostrar por pantalla. De esta manera se conoce el estado del vehículo en todo momento y se puede comprobar si existe algún tipo de problema o si alguno de los comandos no se ejecuta de forma correcta. En la siguiente imagen se puede ver lo que aparece por pantalla al ejecutarse este código:



```
marta@marta-VirtualBox: ~/Documentos/Programas_tfg
marta@marta-VirtualBox: ~
marta@marta-VirtualBox: ~/Documentos/Programas_tfg
marta@marta-VirtualBox: ~/Documentos/ardupilot/ArduCopter$ cd
marta@marta-VirtualBox: ~$ cd Documentos/Programas_tfg
marta@marta-VirtualBox: ~/Documentos/Programas_tfg$ python Proval.py
Connecting to vehicle on: 127.0.0.1:14551
WARNING:dronekit:Link timeout, no heartbeat in last 5 seconds
Basic pre-arm checks
Arming motors
Waiting for arming...
Waiting for arming...
Waiting for arming...
Waiting for arming...
Waiting for arming...
Taking off!
Altitude: 0.002
Altitude: 0.003
Altitude: 0.003
Altitude: 0.43
```

Figura 38: Simulación del programa de despegue y aterrizaje parte 1

```
Altitude: 2.485
Altitude: 4.163
Altitude: 6.452
Altitude: 8.504
Altitude: 9.663
Reached target altitude
Altitude: 9.663000
Altitude: 9.740000
Altitude: 9.793000
Altitude: 9.674000
Altitude: 9.443000
Altitude: 9.193000
Altitude: 8.946000
Altitude: 8.823000
Altitude: 8.580000
Altitude: 8.335000
Altitude: 8.088000
Altitude: 7.839000
Altitude: 7.591000
Altitude: 7.343000
Altitude: 7.094000
Altitude: 6.843000
Altitude: 6.594000
Altitude: 6.346000
Altitude: 6.097000
Altitude: 5.847000
Altitude: 5.722000
Altitude: 5.471000
Altitude: 5.222000
Altitude: 4.974000
Altitude: 4.726000
Altitude: 4.477000
Altitude: 4.229000
Altitude: 3.979000
Altitude: 3.728000
Altitude: 3.479000
Altitude: 3.229000
Altitude: 2.979000
Altitude: 2.729000
Altitude: 2.480000
Altitude: 2.355000
Altitude: 2.106000
Altitude: 1.856000
Altitude: 1.606000
Altitude: 1.358000
Altitude: 1.109000
Altitude: 0.858000
Altitude: 0.607000
LANDED 1
Basic pre-arm checks
Arming Motors
Waiting for arming...
Taking off!
Altitude: 0.006
Altitude: 0.006
Altitude: 0.044
Altitude: 1.087
Altitude: 3.126
Altitude: 5.327
Altitude: 7.025
Altitude: 8.877
Altitude: 9.665
Reached target altitude
Altitude: 9.665000
Altitude: 9.794000
Altitude: 9.793000
Altitude: 9.628000
Altitude: 9.384000
Altitude: 9.258000
Altitude: 9.011000
Altitude: 8.645000
Altitude: 8.522000
Altitude: 8.275000
Altitude: 8.028000
Altitude: 7.780000
Altitude: 7.531000
Altitude: 7.281000
Altitude: 7.030000
Altitude: 6.781000
Altitude: 6.531000
Altitude: 6.407000
Altitude: 6.157000
Altitude: 5.909000
Altitude: 5.659000
Altitude: 5.534000
Altitude: 5.283000
Altitude: 5.034000
Altitude: 4.785000
Altitude: 4.536000
Altitude: 4.286000
Altitude: 4.036000
Altitude: 3.787000
Altitude: 3.536000
Altitude: 3.285000
Altitude: 3.161000
Altitude: 2.911000
Altitude: 2.662000
Altitude: 2.413000
Altitude: 2.164000
Altitude: 1.914000
Altitude: 1.665000
Altitude: 1.415000
Altitude: 1.164000
Altitude: 0.916000
Altitude: 0.667000
LANDED 2
marta@marta-VirtualBox:~/Documentos/Programas_tfg$
```

Figura 39: Simulación del programa de despegue y aterrizaje parte 2

1.3. Vuelo hasta unas coordenadas

Completada con éxito la anterior simulación, el siguiente paso consiste en hacer volar el dron hasta unas coordenadas determinadas, donde debería realizar la entrega de un paquete. El programa que se va a mostrar a continuación consiste en: despegar hasta unos 10 metros de altura aproximadamente y volar hacia unas coordenadas (especificadas en el propio código del programa). Cuando consiga llegar al destino deberá aterrizar y esperar durante 30 segundo, tiempo en el que se realiza la entrega del paquete. Posteriormente despegará y cuando se encuentre a 10 metros de altura volverá al lugar de origen para aterrizar allí nuevamente. En el anexo E.2 se muestra el código del programa que se ha descrito.

Para la simulación de este programa es necesario ejecutar nuevamente el simulador; desde la terminal de Linux se abre la carpeta de ArduPilot y se ejecuta el simulado del multi-roto pero en esta vez es necesario un mapa y una consola donde aparezcan los datos relativos al comportamiento del dron, para ello el comando a utilizar es: `sim_vehicle.py - -console - -map`. La pantalla que se obtendrá al introducir el comando es la siguiente:

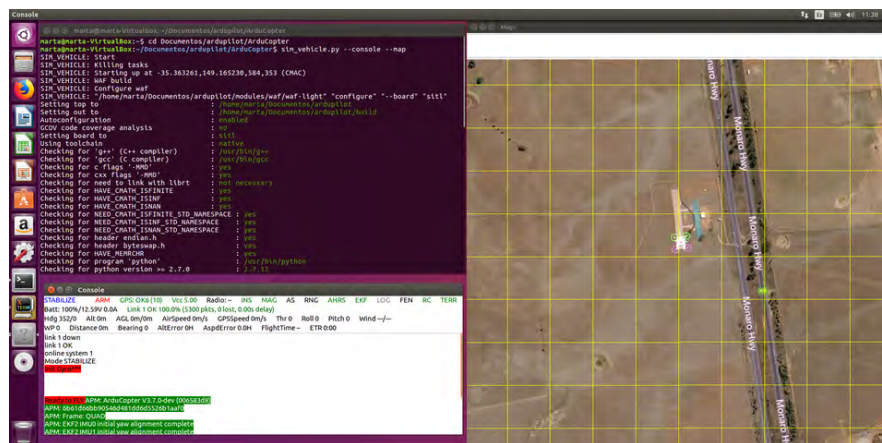


Figura 40: Simulación con mapa y consola

Una vez abierto el entorno de simulación, desde una nueva terminal se abrirá el programa en Python que cuyo código se ha explicado arriba. El ejecutarse aparecerán por la terminal los mensajes que deben mostrarse por pantalla, en la consola se actualizará a tiempo real el estado del vehículo y en el mapa se actualizará, también en tiempo real, la posición en la que este se encuentra. Como en la simulación anterior, el vehículo despegará y por la terminal aparecerán los mismos mensajes. Al llegar a una altura de 10 metros el vehículo se dirigirá a coordenadas especificadas. En la figura 41 se ejemplifica lo que aparece en pantalla en el momento en que el dron se dirige a una ubicación.

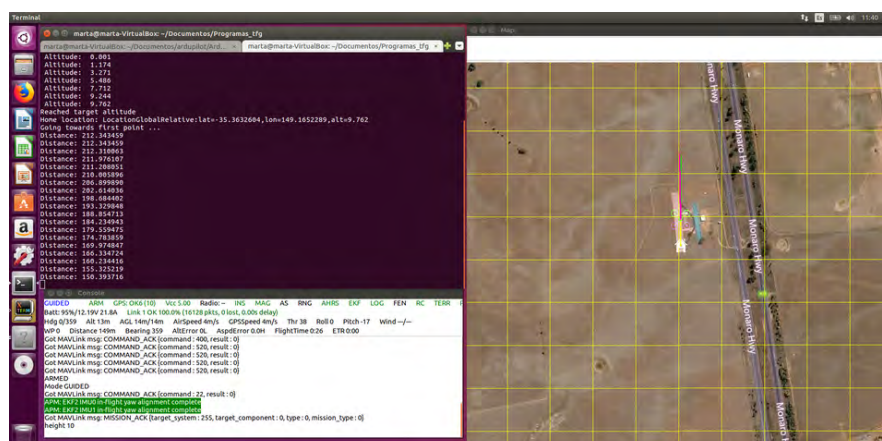


Figura 41: Vehículo dirigiéndose a unas coordenadas

En cuanto a la siguiente imagen, corresponde al instante en que el dron ha llegado a su destino y se dispone a aterrizar. En la terminal se puede ver como la distancia se ha ido reduciendo y al aproximarse al destino empieza el aterrizaje.

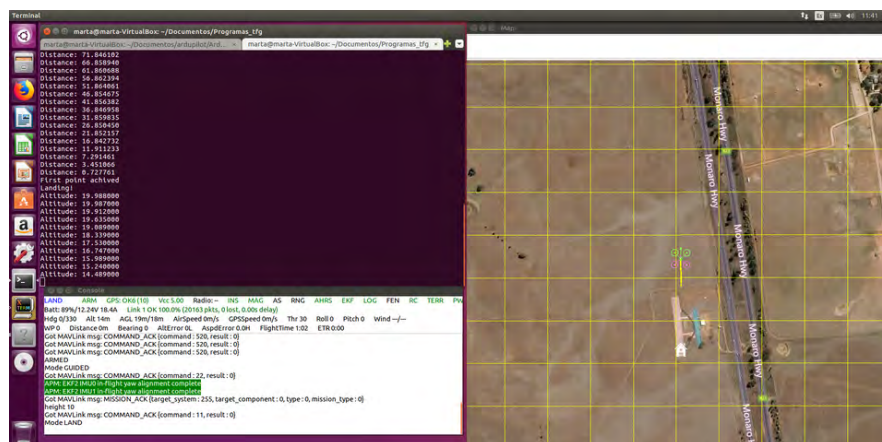


Figura 42: Vehículo llega a su destino y aterriza

Finalmente, se repite el mismo proceso. El dron vuelve a despegar y se dirige al punto desde el que partió. Una vez allí aterrizará para de esta forma finalizar la misión. En la siguiente imagen se muestra como el vehículo consigue aterrizar y se completa con éxito la misión y de esta forma acaba la simulación.

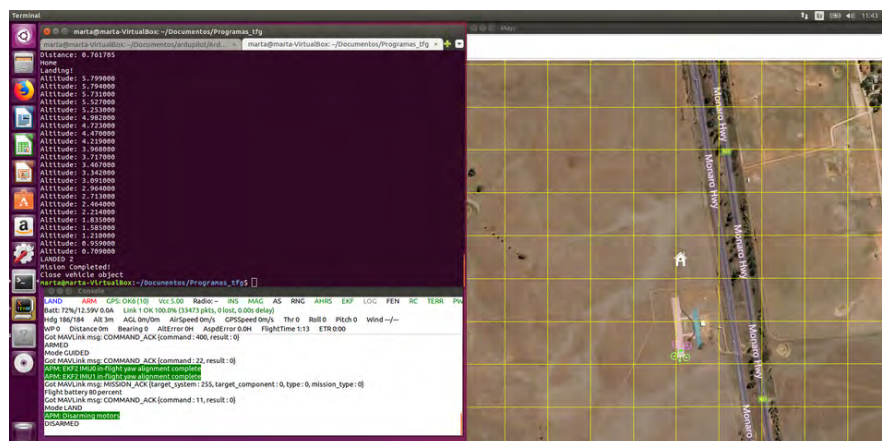


Figura 43: Vehículo llega a casa y se completa la misión

El programa final se basará en este script que ha sido simulado ya que se demuestra que el código es funcional y el vehículo se ha comportado acorde con las funciones programadas y ha cumplido con los requisitos iniciales de este proyecto. Los siguientes pasos se centrarán en adaptar este programa para introducirlo en el prototipo y que cumpla con todas las medidas de seguridad necesarias.

1.4. Cambio de modo por radio control

En esta simulación se va a mostrar como recuperar el control del dron en caso que la misión no se ejecute correctamente. Es decir, si en algún momento se detecta que algo no está funcionando correctamente, el dron se desvía de la ruta, las coordenadas seleccionadas no son las correctas o simplemente se decide abortar la misión; existe la posibilidad de devolver el dron al punto de salida o bien controlarlo desde la unidad de radio control. En el ejemplo que se mostrará a continuación el dron realizará la misma misión que en la simulación anterior, despegará y se dirigirá hasta unas coordenadas para entregar un paquete y posteriormente volverá al inicio, pero, en este caso, se simulará la interrupción de dicha misión. En el anexo E.3 se puede encontrar explicado el código

del programa que se ha descrito, para realizar este programa se ha optado por incorporar algunas funciones de MAVLink que aparecen descritas en la siguiente fuente [49].

Para simular esta misión es necesario conectar dos vehículos. Por un lado debe conectarse con el simulador SITL para poder tener un dron que ejecute todas las funciones y simule el comportamiento de un dron real. Por otro lado, para poder recibir datos desde la unidad de control remoto o mando es necesario realizar una segunda conexión, tal y como se explica en el apartado de conexiones, el receptor debe conectarse a la controladora la cual recibe las ordenes y las ejecuta. Por tanto, el segundo vehículo que se conectará es la controladora Pixhawk 4 que recibirá las ordenes del mando para detener la simulación. En la siguiente figura se muestran las conexiones que se deben realizar.

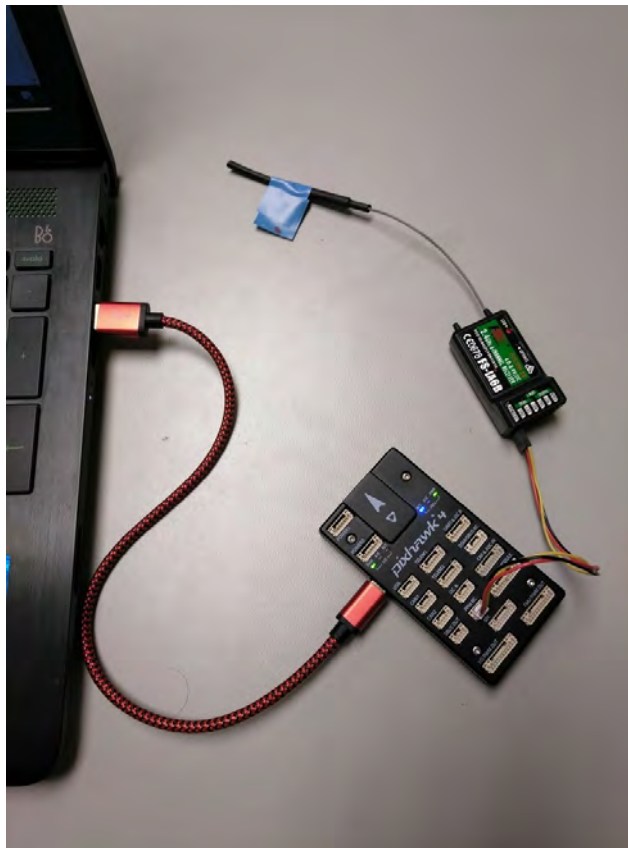


Figura 44: Conexión de la controladora y el receptor

Para realizar la simulación, primero es necesario abrir la estación de control en tierra QGroundControl. En cuanto esté operativa se ejecutará el simulador; desde la terminal de Linux abrir la carpeta de ArduPilot y ejecutar el simulador multi-rotor. Esta vez la simulación se realizará en el campo de vuelo del Club Aeronáutico Egara. Para poder situar la simulación en esta ubicación es necesario realizar una serie de cambios. Siguiendo al ruta Ardupilot/Tools/autotest se encuentra un fichero txt llamado locations que incluye todas las posibles localizaciones en las que se pueden realizar las simulaciones. Para añadir una nueva simplemente se debe escribir en una nueva línea las coordenadas de esta localización, altura respecto al suelo y orientación del vehículo, en este caso sería Egara = 41.574841, 1.997031, 0, 240. Una vez añadida la nueva localización, para que el simulador se pueda ejecutar en ella el comando que se debe escribir en la terminal es el siguiente `sim_vehicle.py -L Egara` [50]. En la pantalla aparecerá lo siguiente:

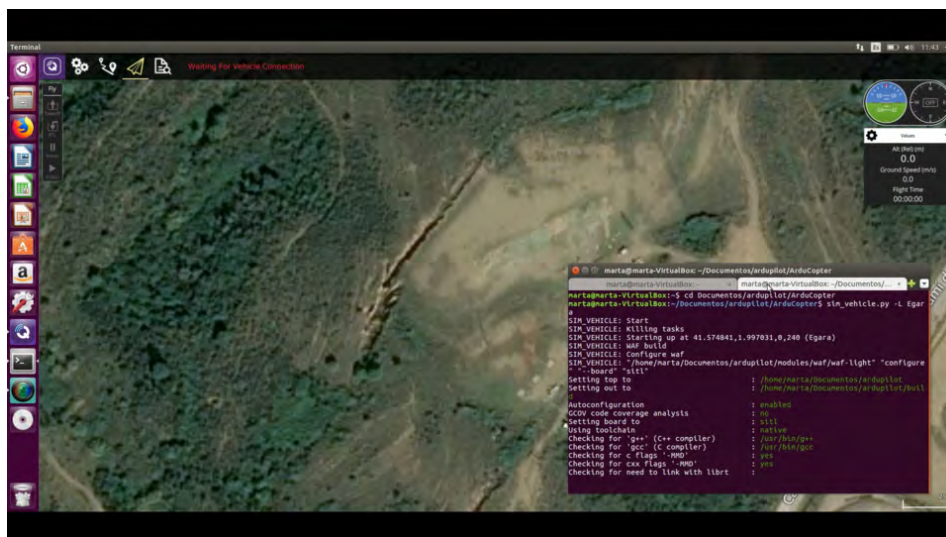


Figura 45: Simulación en Club Aeronáutico Egara

En el momento en que la estación de control se conecte con el vehículo simulado, se realizará la conexión de la controladora Pixhawk 4 al ordenador por medio de USB, de esta forma estarán los dos vehículos conectados al ordenador. Antes de realizar la simulación del programa es necesario encender la unidad transmisora del sistema de radio control (el mando) y situar el interruptor correspondiente al canal 6 en la posición 2, hacia abajo tal y como se muestra en la figura 46. Es importante situar el interruptor en dicha posición ya que, según el código realizado, el programa se detendrá cuando el interruptor esté en la posición 1, por lo tanto para que pueda iniciarse no puede estar en esa posición. Hechas todas las conexiones necesarias, es el momento de simular el programa. Para ello desde la terminal de Linux se abre la carpeta en la que se encuentra el programa y se simula con la orden python seguida del nombre del archivo. En este momento se conectará con ambos vehículos, el simulado y la controladora conectada por USB y se empezarán a ejecutar el código que se ha descrito.



Figura 46: Preparación de la unidad transmisora

Por pantalla aparece en la estación de control QGroundControl la posición del vehículo en cada momento y los datos de altura, velocidad, batería y el modo de vuelo, entre otros. Por otro lado, en la terminal aparecerán los mensajes, que según el código, deben mostrarse por pantalla en cada momento. Del mismo modo que en la simulación anterior el vehículo se ha armado y ha despegado hasta una altura de 10 metros. Una vez alcanzada esa altura se dirige hasta las coordenadas especificadas

en el código, también dentro del campo de vuelo. En la siguiente figura se muestra, el funcionamiento normal de la misión, en concreto, como el vehículo se está dirigiendo a su destino.

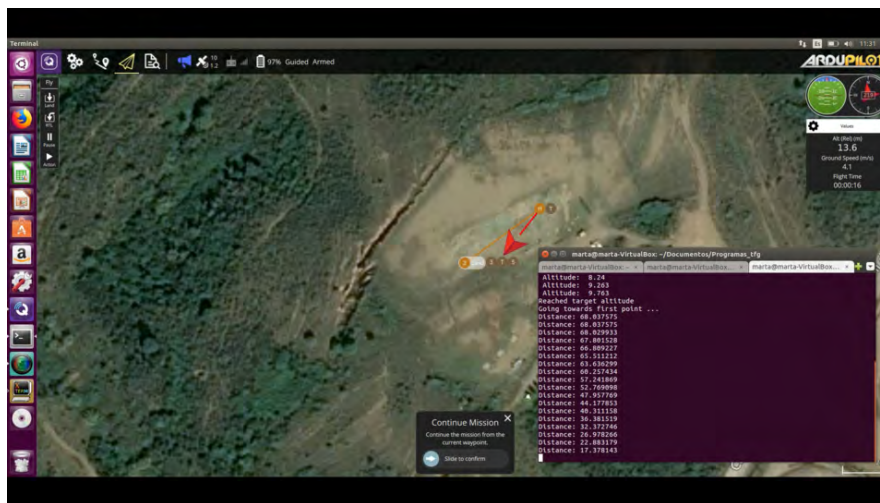


Figura 47: Ejecución de la simulación

En el momento en que el vehículo se dispone a llegar a su destino se ha interrumpido su misión. Para ello, como se ha descrito anteriormente, es necesario situar el interruptor correspondiente al canal 6 en la posición 1, es decir, hacia arriba. En el momento en el que eso sucede el vehículo cambiará de modo y posteriormente saldrá del programa interrumpiendo así las funciones que debía realizar. En este caso el modo elegido es RTL (Return To Launch) ya que en este tipo de simulaciones no es posible obtener el control del vehículo simulado a través de la unidad de radio control ya que se ha conectado como otro vehículo distinto. En la siguiente figura se muestra el cambio de posición del interruptor y por pantalla se puede observar en la terminal de Linux como se ha dejado de ejecutar el programa.



Figura 48: Interrupción de la simulación

Finalmente, el vehículo, según el nuevo modo de vuelo, volverá al lugar desde el cual despegó y una vez allí aterrizará y se desarmará terminando así su vuelo. En la figura se muestra la pantalla después de haber activado el interruptor, se observa como el vehículo ha cambiado el modo de vuelo, en la barra superior aparece el modo de vuelo es RTL y en el mapa se puede ver como el vehículo se está dirigiendo al lugar desde el cual despegó. Además en la terminal se observa que el programa sigue detenido en el mismo lugar y no se han seguido ejecutándose el código después de activar el interruptor.

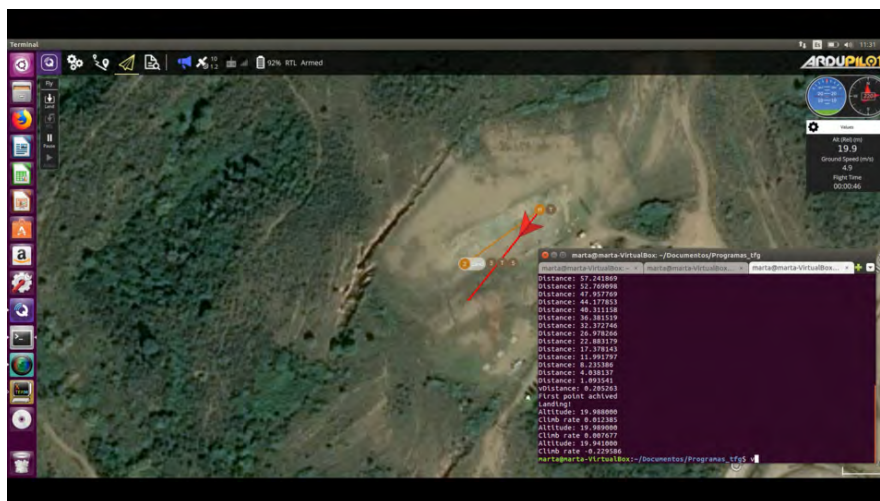


Figura 49: Vehículo cambia de modo y se detiene el programa

En esta simulación se ha mostrado con éxito como detener el programa y tomar el control del dron si sucede algún tipo de problema durante el vuelo. El programa definitivo, por lo que respecta a la función de control, será muy similar al simulado pero esta vez todo conectado en un mismo vehículo.

2. Dispositivos de visibilidad

Los dispositivos de visibilidad seleccionados son programables, por tanto es necesario realizar un programa para encender los led de la forma deseada. Para ello es necesario conectarlos a la Raspberry Pi, que debe ser configurada previamente. Para poder trabajar con ella es necesario instalar el sistema operativo Raspian en una tarjeta micro SD. Una vez hecho esto se introduce la tarjeta en la Raspberry. Para encender la Raspberry basta con darle voltaje, es decir, hay conectarla a la corriente, pero para poder trabajar con ella también será necesario conectarle una pantalla, un teclado y un ratón; de esta forma podremos trabajar con ella como si se tratará de un ordenador de sobremesa.



Figura 50: Puesta a punto del ordenador Raspberry Pi

Por lo que se refiere a los dispositivos de visibilidad es necesario instalar Python3 y una serie de librerías para poder ejecutar el programa que controle las luces. Primero se instalará Python3 ya que las librerías que se utilizarán solo funcionan en esta versión del programa. Una vez instalado se activarán los conectores RPI.GPIO, donde irá conectado el cable de señal de las luces y se instalará la librería

adafruit-blinka y la correspondiente a la tabla de luces elegida `rpi_ws281x` `adafruit-circuitpython-neopixel` [51]. Una vez realizados todos los pasos la Raspberry está lista para ejecutar el programa de control de luces deseado. En cuanto a la conexión, tal y como se especifica en el esquema de conexiones, las luces tienen tres cables: positivo, negativo y señal. El positivo se conectará en uno de los pins de la Raspberry que de 5V, el negativo a tierra y el de señal se en el GPIO18. La conexión de las luces así como la de todos los dispositivos conectados a la Raspberry se muestran en la siguiente figura. También se puede observar que los cables han sido alargados para poder situar cada placa en uno de los brazos del dron y se han adaptado las conexiones para los pines de la Raspberry.

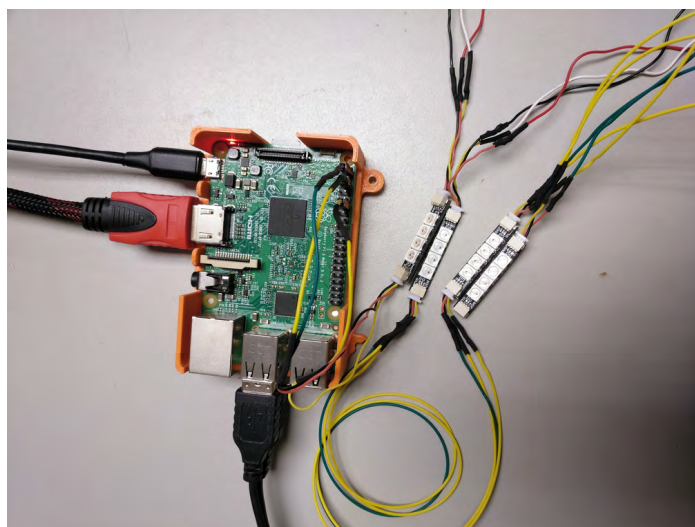


Figura 51: Conexión de luces en Raspberry Pi

Realizadas todas las conexiones e instaladas todas las librerías necesarias es posible realizar el código de encendido de las luces. Tal y como se ha explicando a lo largo del trabajo es importante diferenciar los brazos delanteros y traseros, por tanto el código realizado permite que las dos placas situadas en los brazos delanteros emitan luz verde, mientras que las placas situadas en la parte trasera la emitan roja. Se han elegido estos colores porque son los que consumen menos potencia. Para poder ejecutarlo, en la Raspberry, desde la terminal, se abre la carpeta en la que está situado y se utiliza el siguiente comando: `sudo python3` seguido del nombre del archivo. En el momento en que se ejecute le encenderán las luces de la siguiente forma:

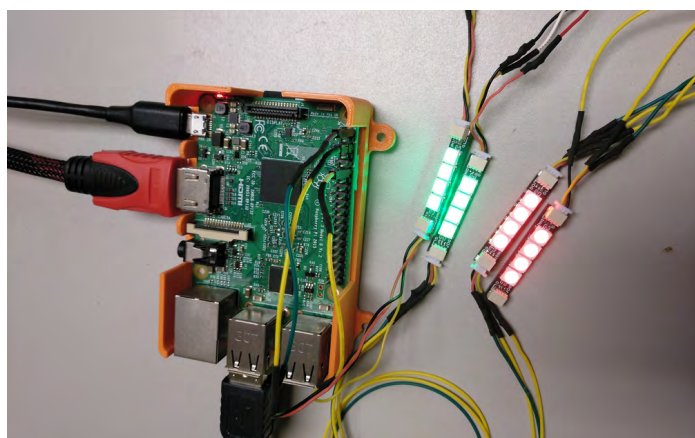


Figura 52: Luces led encendidas

Se ha podido comprobar que el programa de control de las luces funciona correctamente y se ha obtenido el resultado esperado. El código correspondiente a este programa se encuentra comentado en el anexo E.4.

3. Dispositivo de visión

Tal y como se especifica en la normativa, si el dron volara fuera del alcance visual del piloto, este debería tener un dispositivo de visión enfocado hacia delante para poder observar hacia donde se dirige el vehículo. Para ello se ha optado por la grabación de vídeo en streaming con una cámara Raspberry Pi. Esta cámara irá colocada en la parte delantera del vehículo, atornillada a la pieza mostrada en el apartado de prototipo.

Para grabar un vídeo en streaming, lo primero que debe hacerse es habilitar la cámara en el menú de configuración de Raspberry Pi para que el ordenador sea capaz de detectarla una vez se haya conectado. Una vez hecho esto será necesario saber la dirección IP de la Raspberry, para ello basta con introducir en la terminal el comando `ifconfig` y nos aparecerá por pantalla. En este caso la dirección IP es 10.193.210.196. Es importante guardar este número ya que será necesario usarlo más adelante. Seguidamente conectamos la cámara a la placa tal y como se muestra en la figura:

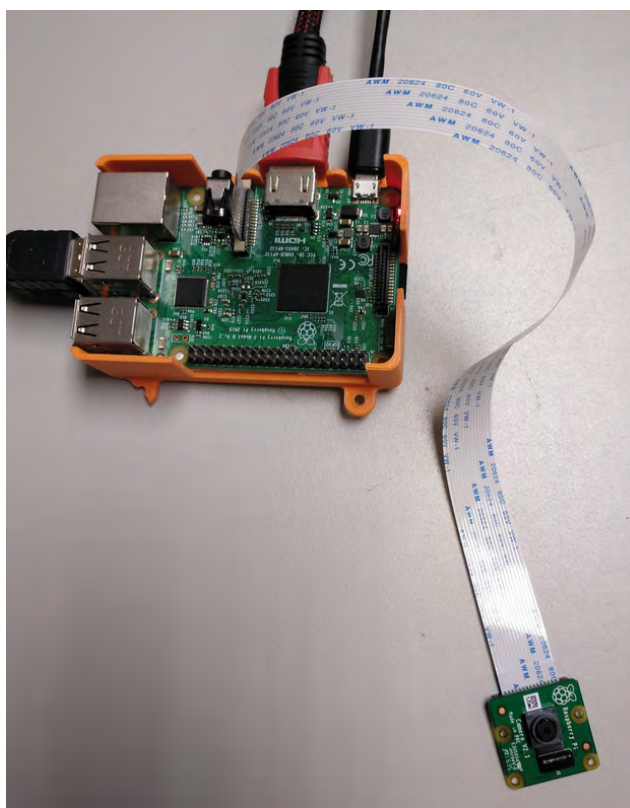


Figura 53: Conexión del modulo cámara

En cuanto a la grabación y visualización del vídeo en directo es necesario un script en Python que nos permita retransmitir ese vídeo y verlo desde una unidad remota. Puede ser visto desde cualquier dispositivo conectado a la misma red Wifi que la Raspberry. Para realizar la grabación simplemente debe ejecutarse el código y, una vez esté corriendo, entrar en el servidor web siguiente: `http://` seguido de la dirección IP correspondiente a la Raspberry y por último `:8000`. Nos aparecerá una pantalla con las imágenes que la cámara está grabando a tiempo real. La información necesaria para realizar este programa se encuentra detallada en la web de Raspberry [52] y código utilizado puede encontrarse en el anexo E.5.

Parte X

Pruebas de vuelo

Comprobado el correcto funcionamiento de los programas realizados, el siguiente paso consistirá en probar estos programas en el prototipo construido para verificar así su eficacia. En este apartado se explicará los pasos seguidos antes y durante las pruebas de vuelo así como el resultado obtenido en cada una de ellas.

1. Estructura

Durante las pruebas de vuelo hay que realizar una serie de comprobaciones para asegurar el correcto funcionamiento de todos los componentes. Es decir, antes de probar en el quadcopter el programa realizado primero es necesario asegurarse que los componentes funcionan correctamente y responden de manera adecuada a las ordenes que reciben. A continuación se detallan las fases en las que se han dividido las pruebas de vuelo para asegurar su correcto desarrollo.

1.1. Comprobaciones previas

El primer paso, antes de ir al campo de vuelo, es la puesta a punto de la controladora y el ordenador de abordó. Para poder ejecutar el programa diseñado es necesario instalar Ardupilot en la controladora. Como se ha explicado en el apartado de entorno de programación, Ardupilot es un software de autopiloto que soporta una gran diversidad de plataformas y vehículos y en este caso ha sido el elegido en el momento de realizar la programación. Por otro lado, también se debe instalar la librería DroneKit en la Raspberry para que sea capaz de ejecutar las funciones del sistema de guiado que se encuentran en el programa.

Con todos los componentes listos y realizado el montaje y conexión del prototipo, el siguiente paso es su calibración. La calibración se realizará con la ayuda de la estación de control elegida, en este caso QGroundControl. Para ello, es necesario conectar la telemetría al ordenador y, con las hélices quitadas, conectar la batería al dron. La estación de control detectará el vehículo conectado y de esta forma podrá iniciarse la calibración de los sensores, en concreto del magnetómetro y giroscopio, pero también será necesario un level horizon para calibrar el horizonte artificial. Por otra parte, hay que realizar la configuración de la unidad de radio control, asegurando así que todos los comandos funcionan correctamente y probar los distintos canales.

Finalizada la puesta a punto del prototipo, se llevará a cabo una última comprobación. Con las hélices quitadas, se probará el correcto funcionamiento de los sensores calibrados. Desde la unidad de control se dará un poco de aceleración (throttle) a los motores para ver como se comportan y se orientará el vehículo en distintas posiciones para comprobar que el giro de los motores es el correcto para intentar volver a poner el dron en la posición original y contrarrestar así la perturbación dada. También se puede comprobar si el horizonte artificial marca la posición real del vehículo. Si todas las comprobaciones explicadas han dado un resultado satisfactorio, el vehículo está listo para volar.

1.2. Vuelo con unidad de radio control

Después de asegurar que el prototipo funciona correctamente en tierra y sin hélices, es momento de probar su comportamiento en el aire. Antes de realizar cualquier misión es importante comprobar que puede volar de forma estable y que responde de forma correcta a las ordenes recibidas. Para ello, se realizará un vuelo de prueba con la unidad de radio control.

En este primer vuelo se probarán los cuatro elementos fundamentales para el control del dron: aceleración, cabeceo, guiñada y alabeo. Es importante comprobar que ninguna de estas ordenes se haya configurado del revés, es decir, que el dron haga la guiñada hacia la izquierda cuando desde el mando

se le ordene hacerla a la derecha. El buen funcionamiento de la unidad de control remoto es clave ya que si sucede algún problema durante la misión se tomará el control del vehículo a través de ella para asegurar de nuevo su estabilidad y aterrizarlo de forma segura.

Obtenida una buena respuesta de los controles, el siguiente paso consiste en probar los distintos modos de vuelo. Los más habituales son: *stabilize*, con el que se han realizado las comprobaciones anteriores; *guided*, utilizado en los programas realizados; *altitude hold*, se mantendrá constante la altura a la que está volando el dron; y *position hold*, se mantendrá el dron en la posición en la que se encuentra en el momento de activar este modo. Los modos de vuelo se pueden cambiar a través de los canales de la unidad de control, estos deben definirse previamente con la ayuda de la estación de control. Superada con éxito la prueba de vuelo con radio control, se pueden empezar a realizar misiones.

1.3. Vuelo con QGroundControl

Antes de realizar una misión hay que configurar los canales del mando para poder cambiar de modo de vuelo si se detecta algún problema. Los canales 5 y 6 están asociados a los interruptores del mando, estos pueden tener dos o tres posiciones, de manera que se pueden configurar hasta tres modos de vuelo. La misiones se llevan a cabo en modo auto, para tomar el control de la aeronave se deberá cambiar a *stabilize*. El tercer modo puede ser *position hold* o *altitude hold*, modos que estabilizarán el vehículo antes de coger el control. Para realizar el cambio de modo simplemente hay que mover el interruptor hasta la posición correspondiente al modo deseado.

Para construir una misión con QGroundControl, hay que situarse en el mapa y marcar los puntos por los que se desea que el dron pase. También hay que señalar que se lo que debe hacer el dron en ese punto, es decir, cuando el dron llegue a las coordenadas propuestas deberá aterrizar en ellas, después despegar y por último volver al origen. En el mapa se muestran todas las ordenes programadas y durante el vuelo se actualiza el estado y posición del vehículo a tiempo real. Esta prueba de vuelo es necesaria para comprobar el correcto funcionamiento del GPS y la actitud del dron cuando vuela de forma autónoma. Si se consigue un buen comportamiento en este tipo de misiones el prototipo estará listo para probar los programas en Python y comunicarse con el ordenador de abordo.

1.4. Vuelo con el programa

Por último, se probarán los programas realizados y simulados anteriormente. Antes de eso se debe comprobar la correcta comunicación entre la controladora y el ordenador de abordo ya que las ordenes en este caso se recibirán a través de este ordenador. También es necesaria la conexión entre los dos ordenadores, el portátil utilizado como estación de control y la Raspberry que realiza la función de ordenador de abordo. Para ello ambos dispositivos tienen que estar conectados a una misma red. Su conexión debe hacerse desde la terminal de linux del ordenador portátil, hay que ejecutar el comando `ssh pi@` seguido de la dirección de la Raspberry. De esta forma desde el portátil se podrá tomar el control de la Raspberry y acceder a su terminal para ejecutar así los programas que contenga.

Es importante asegura que el comando de recuperación con la unidad de radio control funciona correctamente. Es decir, en el código se describe que cambiando de posición uno de los canales del mando (en los anexos E.6 y E.7 se especifica cual es) el programa va a ser interrumpido, el modo de vuelo cambiará a *stabilize* y se saldrá del código. De esta forma pasará a tenerse el control total del dron a través la unidad de radio control. Si esta función se realiza correctamente se puede proceder a ejecutar el programa desarrollado a lo largo de este proyecto.

2. Vuelos

Las pruebas de vuelo han seguido la estructura explicada anteriormente. Para poder llevarlas a cabo es necesario contar con alguien que sea capaz de pilotar el dron, por este motivo las pruebas de vuelo se han realizado con la ayuda de los miembros del proyecto Aldora. El lugar en que se han hecho ha sido el Club Aeronáutico Egara¹. Se han realizado principalmente tres pruebas de vuelo, en dichas pruebas de vuelo han surgido diversos problemas relacionados mayoritariamente con la controladora. Es por eso que estas en cada una de ellas se ha utilizado una controladora distinta para intentar encontrar unas prestaciones óptimas durante el vuelo. A continuación se detallarán en que han consistido las pruebas de vuelo, sus resultados y conclusiones. En las pruebas de vuelo no se ha añadido el dispositivo de reducción de energía en caso de caída, es decir, no incorporaba el paracaídas.

2.1. Vuelo 1: Pixhawk 4

La primera prueba de vuelo se realizó con el prototipo diseñado y con todos los componentes que se especifican a lo largo del trabajo. Realizadas las comprobaciones previas, se hicieron las primeras pruebas con la unidad de radio control. Los controles de aceleración, alabeo, guiñada y cabeceo respondieron correctamente. En cuanto a los modos de vuelo, en altitude hold el comportamiento mostrado fue bastante bueno y en position hold mostraba dificultades para mantener la posición. Estos problemas de precisión podrían ser debidos a los valores dados al PID de la controladora.

Pese a no obtener unos resultados óptimos, se procedió a probar de realizar una misión con QGround-Control. Primero se optó por una misión sencilla que consistía solamente en despegar, volar hacia unas coordenadas determinadas señaladas en el mapa, a unos 20 metros del origen, y aterrizar allí.



Figura 54: Primer vuelo hacia unas coordenadas

Durante el transcurso de la misión se pudieron observar las dificultades experimentó el quadcopter para llegar al punto señalado en el mapa a causa de la falta de precisión ya mostrada en las pruebas con

¹Este campo de vuelo está situado a las afueras de Terrassa, en la Avenida de Can Boada

radio control. Pero, una vez alcanzada la posición indicada, realizó el aterrizaje de forma correcta y dentro de los márgenes de error establecidos. La misión se completó con éxito sin necesidad de intervención.



Figura 55: Primera prueba finalizada

Para asegurar el buen comportamiento del prototipo se realizó otra misión, esta vez completa. Debía, como en la anterior, despegar y volar hasta las mismas coordenadas para realizar un aterrizaje allí; después de aterrizar mantenerse en tierra unos 30 segundos para volver a despegar y regresar al lugar desde el cual despegó la primera vez. La primera parte la realizó correctamente pero al volver un problema relacionado con el GPS hizo que intentará corregir su posición y se alejó mucho la teórica ruta de regreso. Como consecuencia de ello se tubo que intervenir y tomar el control a través del mando. Los problemas con el GPS impidieron pasar a la fase final y probar los programas realizados.

2.2. Vuelo 2: Pixhawk 1

Ante los problemas mostrados en la prueba de vuelo anterior, se concluyó que el error no era un problema del GPS sino de la lectura de datos que había hecho la controladora. Como solución se optó por realizar más pruebas con otra controladora, la Pixhawk 1, para ver si esta se comportaba mejor y era más precisa que la anterior. Para ello simplemente se cambió la pieza de soporte de la controladora por otra que se adaptada a la forma de la nueva controladora elegida.

Durante las pruebas con la unidad de radio control surgieron muchos problemas. La controladora no era capaz de mantener un vuelo estable. No respondía correctamente a los controles básicos de aceleración, cabeceo, alabeo y guiñada. Solo con acelerar un poco los motores cogían unas velocidades de giro muy altas y el vehículo se elevaba mucho haciendo imposible su control. Además, no era posible controlar su guiñada.



Figura 56: Comportamiento inestable del prototipo

Estos problemas de control derivaron en una caída desde varios metros del suelo y como consecuencia de esta en la rotura del prototipo. Aún y así, con la ayuda de un soldador y cinta adhesiva se pudo reconstruir para realizar alguna prueba más e intentar solucionar los problemas de control con otra calibración y tocando parámetros desde la estación de control. No fue posible descubrir donde residía el problema y en las siguientes pruebas de vuelo no se obtuvieron mejores resultados. Por lo tanto, en esta segunda prueba de vuelo no se pudo probar ninguna misión, ni desde QGroundControl, ni a partir de los programas realizados.

2.3. Vuelo 3: Pixhawk 2 (Cube)

En un intento de mejorar los resultados obtenidos en la primera prueba de vuelo y con el prototipo en muy mal estado, la tercera prueba de vuelo se realizará con una versión del prototipo que dispone de los mismos componentes elegidos, a excepción de los dispositivos de visibilidad que no los lleva incorporados. La controladora es una Pixhawk 2 y el GPS, que quedó dañado de la prueba de vuelo anterior, también ha sido substituido por el que incluye el kit de esta nueva controladora. Se ha reducido el número de piezas impresas en 3D para simplificar el montaje, tanto la antena de comunicación como el receptor están fijados en los brazos del quadcopter con bridas. El prototipo utilizado es el que se muestra en la siguiente figura:



Figura 57: Prototipo tercera prueba

La prueba con la unidad de radio control dio unos resultados excelentes tanto en los controles como en los modos de vuelo. Además se probó con éxito el modo RTL (Return To Land), cuando la aeronave se encuentra a una cierta distancia del lugar de origen se activa este modo que consiste en devolverla hasta la ubicación desde donde despegó, a una altura preestablecida (15 metros), y aterrizar allí de forma completamente autónoma.

Debido a los buenos resultados obtenidos en el fase anterior se prosiguió a realizar una misión desde QGroundControl. La misión que se muestra en la figura inferior consistía en volar hasta el punto numero 2 y aguantar 10 segundos la posición, ir hasta el punto 3 y allí aterrizar. Después de una espera de 30 segundos en tierra volver a despegar y regresar al lugar de origen. Esta misión se llevó a cabo en tres ocasiones y en las tres se obtuvieron resultados satisfactorios.

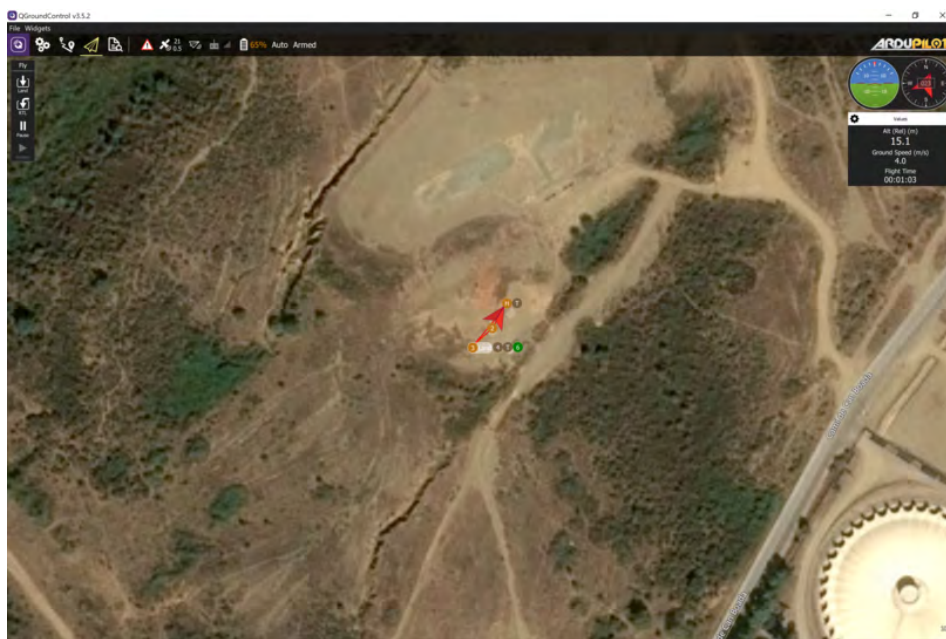


Figura 58: Misión con QGroundControl

En este punto se dan las condiciones necesarias para probar los programas realizados. Para ello es necesario configurar la unidad de control de acuerdo con las condiciones impuestas en los códigos que se encuentran en los anexos E.6 y E.7. Para mayor seguridad se ha optado por armar el vehículo desde el mando, la condición para que el código pueda empezar a ejecutarse es situar el canal 5 en la segunda posición, posición reservada para el modo guided. Como se ha explicado anteriormente la posición 3 de este mismo canal sirve para interrumpir el código y poner el vehículo en modo stabilize. Por tanto en la posición 1 se podrá configurar el modo en que se desee y en el cual se armará el vehículo, en este caso position hold.

Para empezar se probará un código sencillo para comprobar que el vehículo se comporta correctamente y que es posible detener el programa en cualquier momento. Este primer código, que se encuentra comentado en el anexo E.6, consiste en un simple despegue hasta 10 metros de altura y una vez alcanzados procederá al aterrizaje en ese mismo punto. El código es sencillo y los resultados obtenidos han sido excelentes. En la siguiente figura se puede observar los mensajes que aparecían por el terminal de la Raspberry mientras se llevaba a cabo la misión.

```

pi@raspberrypi: /media/pi/R2-D2/TFG/Proves_vol
Waiting for arming...
Waiting for arming...
CRITICAL:autopilot:PreArm: Hardware safety switch
Waiting for arming...
Waiting for arming...
Waiting for arming...
Waiting for arming...
pi@raspberrypi: /media/pi/R2-D2/TFG/Proves_vol $ nano takeoff_land_simple.py
pi@raspberrypi: /media/pi/R2-D2/TFG/Proves_vol $ python takeoff_land_simple.py
Connecting to vehicle on: /dev/ttyACM0
CRITICAL:autopilot:PreArm: Hardware safety switch
Waiting for arming...
Waiting for arming...
Waiting for arming...
Waiting for arming...
Waiting for arming...
Waiting for arming...
Waiting for arming...
Waiting for arming...
Waiting for arming...
Waiting for arming...
Waiting for arming...
Taking off!
Altitude: -0.017
Altitude: -0.039
Altitude: 0.477
Altitude: 2.388
Altitude: 4.622
Altitude: 6.927
Altitude: 8.872
Altitude: 9.561
Reached target altitude
Landing!
Altitude: 9.561000
Climb rate 0.438443
Altitude: 9.780000
Climb rate 0.096120
Altitude: 9.806000
Climb rate -0.106530
Altitude: 9.610000
Climb rate -0.434091
Altitude: 9.391000
Climb rate -0.435453
Altitude: 9.147000
Climb rate -0.468265
Altitude: 8.915000
Climb rate -0.557900
Altitude: 8.703000
Climb rate -0.443038
Altitude: 8.472000
Climb rate -0.614135
Altitude: 8.153000
Climb rate -0.606920
Altitude: 7.868000
Climb rate -0.547819
Altitude: 7.615000
Climb rate -0.535285
Altitude: 7.487000
Climb rate -0.450886
Altitude: 7.254000
Climb rate -0.564210
Altitude: 6.989000
Climb rate -0.532419
Altitude: 6.715000
Climb rate -0.508523
Altitude: 6.448000
Climb rate -0.538348
Altitude: 6.152000
Climb rate -0.505919
Altitude: 5.852000
Climb rate -0.463021
Altitude: 5.594000
Climb rate -0.485208
Altitude: 5.384000
Climb rate -0.439387
Altitude: 5.017000
Climb rate -0.398417
Altitude: 4.893000
Climb rate -0.386685
Altitude: 4.584000
Climb rate -0.275872
Altitude: 4.295000
Climb rate -0.348833
Altitude: 4.061000
Climb rate -0.345925
Altitude: 3.825000
Climb rate -0.395080
Altitude: 3.578000
Climb rate -0.431894
Altitude: 3.335000
Climb rate -0.447627
Altitude: 3.084000
Climb rate -0.455744
Altitude: 2.830000
Climb rate -0.492244
Altitude: 2.540000
Climb rate -0.477037
Altitude: 2.250000
Climb rate -0.415084
Altitude: 1.968000
Climb rate -0.390081
Altitude: 1.840000
Climb rate -0.350432
Altitude: 1.550000
Climb rate -0.321115
Altitude: 1.267000
Climb rate -0.346183
Altitude: 0.994000
Climb rate -0.321996
LANDED
close vehicle object
pi@raspberrypi: /media/pi/R2-D2/TFG/Proves_vol $

```

Figura 59: Comunicación con Raspberry en vuelo

La última y definitiva prueba de vuelo consiste en la ejecución del código completo. Este código se encuentra explicado en el anexo E.7 y consiste en realizar una misión como la que se ha realizado anteriormente con QGroundControl: despegar, volar hasta esas mismas coordenadas, aterrizar y esperar en tierra 30 segundos. Posteriormente despegar y volver al lugar de origen para aterrizar en él de nuevo. Durante esta misión se tendrá además un dispositivo de visión enfocado hacia delante para poder observar hacia donde se está dirigiendo el dron. A continuación se muestran una recopilación de imágenes del quadcopter realizando la misión. La figura 60 se puede ver el inicio de la misión, en el que el dron realiza el despegue hasta una altura de 10 metros. También se observa al piloto pendiente del comportamiento del vehículo y con el mando preparado para intervenir si fuera necesario. Además, la ingeniera a cargo del proyecto se encuentra pendiente de la estación de control en tierra en la que se muestra el estado del dron y las comunicaciones con la Raspberry.



Figura 60: Despegue desde el programa

Como se ha detallado anteriormente, el quadcopter incorpora una cámara enfocada hacia delante que graba durante el vuelo. La siguiente figura corresponde a la visión que tiene el dron en su vuelo hacia las coordenadas especificadas en el código. En ella se pueden ver las hélices del dron y una vista de la ciudad desde 10 metros de altura. La imagen está sacada del vídeo que realizó la cámara Raspberry. Lamentablemente la calidad de la imagen no es óptima y se observa un poco oscura probablemente por la posición del sol en el momento del vuelo.



Figura 61: Visión del vuelo desde el quadcopter

Cuando el vehículo llega a su destino, se ha dado un margen de error menor a 0'5 metros, realiza la maniobra de aterrizaje sobre ese punto. En la figura inferior se puede observar esa maniobra y de nuevo aparece el piloto vigilando que se ejecute de forma correcta por si es necesario tomar el control del quadcopter. Una vez en el suelo realizará una espera de 30 segundos, tiempo para depositar un paquete, y transcurrido ese tiempo despegará de nuevo hasta una altura de 10 metros.



Figura 62: Llegada al destino y aterrizaje autónomo

Finalmente, tras realizar con éxito la misión, el dron regresa al punto de partida, como en el caso anterior el margen de error hasta ese punto es menor a 0'5 metros. Cuando logre situarse en esa ubicación aterrizará completando así la misión programada. En la figura se observa la maniobra de aterrizaje final en que el dron está a 1 metro aproximadamente del suelo. El prototipo ha sido capaz de realizar con éxito la misión, sin necesidad de intervención por parte del piloto durante el transcurso de esta.



Figura 63: Regreso al lugar de partida y fin de la misión

Después de esta prueba de vuelo se puede demostrar la eficacia del programa realizado puesto que el prototipo ha sido capaz de ejecutar todas las funciones que aparecen en el programa con un buen comportamiento y sin necesidad de intervención. También es posible concluir que la controladora con la que se han obtenido mejores resultados y, por tanto, la óptima para esta aplicación es la Pixhawk 2, en el anexo F se encuentra la ficha técnica de esta controladora.

Parte XI

Resultados

Una vez realizado el trabajo, se hará una valoración de ciertos aspectos como el coste de todo el proyecto una vez construido el prototipo, las implicaciones ambientales que tendría si se implementará a nivel comercial esta nueva forma de reparto, como se han planificado y programado las distintas subtarefas en las que se divide el proyecto durante los meses en los que se ha llevado a cabo y, por último, se valorará el resultado final así como las posibles mejoras y aspectos que podrían ser desarrollados en un futuro partiendo de lo que se ha conseguido con este trabajo.

1. Resumen económico

Tal y como se detalla en el documento referente al presupuesto, este ha sido dividido en tres grandes bloques. Por un lado se ha contabilizado el coste de los materiales necesarios para realizar el prototipo. Se ha tenido en cuenta tanto el precio de los componentes seleccionados en el apartado de elección de componentes como los materiales para realizar el chasis y las uniones entre los componentes y el propio chasis. Por otro lado, se ha contabilizado el precio de la electricidad consumida para la realización del proyecto. En él se incluye el consumo de la maquinaria utilizada para llevar a cabo las distintas tareas, tanto la búsqueda de información y redacción como la construcción del prototipo. Finalmente, se han calculado los honorarios de la ingeniera que ha realizado el proyecto teniendo en cuenta las horas dedicadas a su realización. En la tabla que se muestra a continuación se puede observar a cuanto asciende el presupuesto final de el proyecto. Para una información más detallada del presupuesto de cada uno de los bloques, consultar el documento dedicado al cálculo de presupuesto.

Presupuesto parcial	Coste(€)
Materiales	770
Electricidad	40
Recursos humanos	12.400
Total	13.210

Tabla 6: Presupuesto final

2. Implicaciones ambientales

A lo largo del proyecto se ha resaltado que realizar el transporte de mercancías por medio de drones supondría reducir en gran medida la contaminación. Actualmente, los repartos se realizan por medio de furgonetas o camiones de reparto, mayoritariamente propulsadas por un motor de gasolina o Diesel, cosa que además de suponer un gran aumento de los niveles de contaminación también contribuye a aumentar el tráfico. A día de hoy, los niveles de contaminación en las grandes ciudades es alarmante y muchas de ellas empezarán a aplicar restricciones en los días de máximo riesgo. El cambio en la manera en que las mercancías son transportadas puede ser parte de la solución a este gran problema al cual nos enfrentamos.

Poniendo el ejemplo de Amazon, los paquetes de menos de 3 kilos, que son los que potencialmente podrían ser transportados en dron, suponen entre un 80 y un 90 % de las entregas que realiza la compañía, según los datos que parecen en su web [9]. Si estas entregas se hicieran por medio de drones, medios de transporte limpios propulsados por energía eléctrica, supondría una gran reducción de los gases contaminantes. También implicaría una disminución del tráfico, facilitando así la movilidad de los ciudadanos y, lo que es más importante, evitando accidentes causados por los, comúnmente, ajustados plazos de entrega.

3. Planificación y programación

Respecto a la planificación inicial que se hizo del proyecto se ha cumplido casi por completo. Se han reprogramado las tareas de elección componentes, en la que se seleccionaron los sensores y actuadores al mismo tiempo ya que son dos tareas que están fuertemente relacionadas. En la este documento también aparecen descritas en un mismo apartado. También se ha añadido la tarea de simulaciones, que no se describió propiamente como tarea en el Project Charter, fue contemplada dentro de la programación de la función de control por en este caso se considera oportuno asignarle un tarea propia debido al gran numero de horas que se la ha dedicado. Por último, las pruebas de vuelo se han alargado más de lo previsto ya que pretendían acabarse una semana antes de la entrega definitiva pero no ha sido posible en busca de mejores resultados. A continuación, se muestra el diagrama de Gant con la programación definitiva y las fechas actualizadas en las que se han desarrollado todas las tareas en las que subdivide el proyecto realizado.

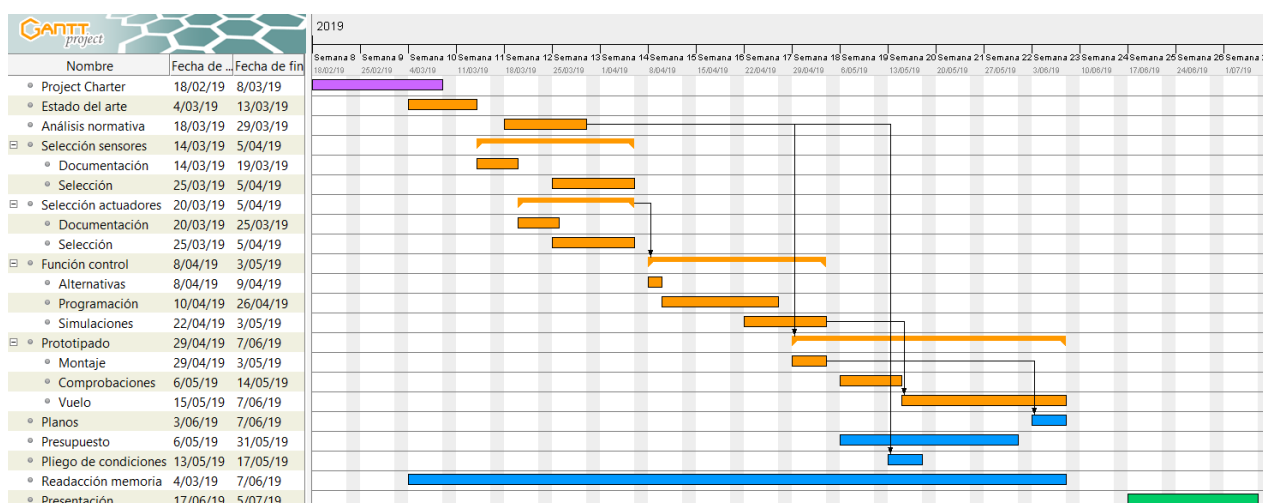


Figura 64: Diagrama de Gant

4. Conclusiones y trabajos futuros

El objetivo de este trabajo era diseñar una función de control que se encargará del sistema de guiado de un dron de servicio de transporte de mercancías para poder realizar el reparto de paquetes hasta un domicilio concreto. Durante el desarrollo del mismo se han seleccionado los sensores y actuadores necesarios, tanto para que el dron pueda volar, como para que pueda hacerlo mediante la función programada y además cumpliendo (en la medida de lo posible) con la normativa española. Los componentes seleccionados se han montado en un dron ya existente y perteneciente al proyecto Aldora, un grupo que forma parte de UPC Space Program, para poder probar en él la función de control programada. Se han desarrollado varios programas que realizan funciones sencillas para poder detectar posibles errores más fácilmente y han sido validados en simulaciones. Cuando el resultado de estas ha sido satisfactorio se ha procedido a la realización de un código final que cumpla con los requisitos establecidos al inicio del trabajo. Finalmente, se han realizado pruebas de vuelo que han demostrado la eficacia del programa realizado.

Hoy en día en España no es posible realizar el transporte por medio de drones, puesto que la normativa vigente no permite transportar mercancías y, además, exige que el vehículo debe estar dentro del alcance visual del piloto o un observador, esto último también dificulta poder llevarlo a cabo. Actualmente, la tecnología avanza muy deprisa y las normativas se van adaptando a las nuevas necesidades, con lo cual es cuestión de tiempo que este medio de transporte sea legalizado. Este trabajo se ha realizado con la visión de que es necesario primero desarrollar la tecnología que lo haga posible con todas las medidas de seguridad necesarias y una vez probada su eficacia y estudiados sus ventajas e inconvenientes pueda ser una realidad en nuestro día a día.

Por lo que respecta a los componentes seleccionados y al prototipo construido, se ha podido ver durante las pruebas de vuelo la importancia de la controladora. En el apartado de elección de componentes se eligió la Pixhawk 4 por ser una de las versiones más recientes de esta marca de controladoras pero la que mejores resultados ha dado durante las pruebas de vuelo ha sido la Pixhawk 2, más testada y en consecuencia con menos errores y un software ya mejorado. Se puede concluir, que el prototipo debe incorporar la Pixhawk 2. El resto de componentes se han comportado de la forma esperada durante los vuelos validado así su elección.

En cuanto a la función de control se ha demostrado que es funcional y el comportamiento del prototipo ha sido el esperado durante el vuelo. Partiendo de este programa se pueden implementar ciertas mejoras para futuros trabajos. Por ejemplo, podría desarrollarse una función para detectar y evitar los obstáculos que pueda encontrarse el dron durante el vuelo. Actualmente el vuelo hacia las coordenadas señaladas lo realiza en línea recta sin tener en cuenta lo que pueda encontrarse en el camino. También se podría trabajar para mejorar el aterrizaje del dron, es decir, que pueda decidir el lugar óptimo para realizar el aterrizaje en vez de el margen de error de 0'5 metros que contiene el programa realizado. Estas son algunas de las principales líneas de trabajo de las empresas del sector que se han comentado en el estado del arte y que actualmente trabajan en mejorar esta tecnología para que sea una realidad en el menor tiempo posible.

Como conclusión, el resultado final de este trabajo ha sido positivo, se han cumplido los requisitos iniciales pese a las dificultades encontradas en las pruebas de vuelo. Se ha diseñado un programa que permite llevar el dron hasta unas coordenadas previamente especificadas y que este aterrice en ellas para hacer una entrega y sea capaz de volver al lugar desde el que despegó. Además, se ha podido probar en un prototipo que ha demostrado ser funcional.

Referencias

- [1] Uso de drones en transporte de mercancías - TomTom Telematics. https://telematics.tomtom.com/es_es/webfleet/blog/efectos-del-uso-de-drones-para-el-transporte-de-mercancias/, Visitado: 2019-02-19.
- [2] Transporte y reparto de mercancías con drones. <https://www.pilotando.es/transporte-y-reparto-de-mercancias-con-drones/#comments>, Visitado: 2019-02-19.
- [3] This was the first FAA-approved autonomous drone delivery in the U.S. | ZDNet. <https://www.zdnet.com/article/this-was-the-first-faa-approved-autonomous-drone-delivery-in-the-u-s/>, Visitado: 2019-02-20.
- [4] Zipline's Ambitious Medical Drone Delivery in Africa - MIT Technology Review. <https://www.technologyreview.com/s/608034/blood-from-the-sky-ziplines-ambitious-medical-drone-delivery-in-africa/>, Visitado: 2019-02-20.
- [5] Watch Flirtey and REMSA deliver defibrillators by drone to 911 callers. <https://www.cnbc.com/2017/10/10/watch-flirtey-and-remsa-deliver-defibrillators-by-drone-to-911-callers.html>, Visitado: 2019-02-20.
- [6] Domino's partners with Flirtey for pizza drone delivery | ZDNet. <https://www.zdnet.com/article/dominos-partners-with-flirtey-for-pizza-drone-delivery/>, Visitado: 2019-02-20.
- [7] Marriott Group teams up with drone pioneers DJI to reinvent the guest experience - TOPHOTELNEWS. <https://tophotel.news/marriott-group-teams-up-with-drone-pioneers-dji-to-reinvent-the-guest-experience/>, Visitado: 2019-02-20.
- [8] Drones Going Postal - A Summary of Postal Service Delivery Drones. <http://unmannedcargo.org/drones-going-postal-summary-postal-service-delivery-drone-trials/>, Visitado: 2019-02-20.
- [9] Amazon.com: Prime Air. <https://www.amazon.com/Amazon-Prime-Air/b?ie=UTF8&node=8037720011>, Visitado: 2019-02-25.
- [10] Drone delivery and parcelcopter technology | Discover DHL. <https://discover.dhl.com/business/business-ethics/parcelcopter-drone-technology>, Visitado: 2019-02-26.
- [11] Deutsche Post DHL Group | DHL Parcelcopter. <https://www.dpdhl.com/en/media-relations/specials/dhl-parcelcopter.html>, Visitado: 2019-02-26.
- [12] X - Wing. <https://x.company/projects/wing/>, Visitado: 2019-03-05.
- [13] Drone meets delivery truck | UPS - Estados Unidos. <https://www.ups.com/us/es/services/knowledge-center/article.page?name=drone-meets-delivery-truck&kid=cd18bdc2>, Visitado: 2019-03-05.
- [14] Van for a highly efficient logistics concept | Daimler > Innovation > Specials > Vision Van. <https://www.daimler.com/innovation/specials/vision-van/en/>, Visitado: 2019-03-05.
- [15] Boeing: Heavy lifter: Boeing reveals new design concept of Cargo Air Vehicle. <https://www.boeing.com/features/2018/07/cav-concept-07-17.page>, Visitado: 2019-03-04.
- [16] Boeing unveiled a prototype for an electric delivery drone — Quartz. <https://qz.com/1177632/boeing-unveiled-a-prototype-for-an-electric-delivery-drone/>, Visitado: 2019-03-04.

- [17] Accelerometer Sensor | MEMS Accelerometer | 3 Axis Accelometer | Arrow.com. <https://www.arrow.com/en/categories/sensors/accelerometers>, Visitado: 2019-03-17.
- [18] Gyro sensors - How they work and what's ahead | about Gyro sensor | Technical Information | other Information | Epson crystal device. https://www5.epsondevice.com/en/information/technical_info/gyro/, Visitado: 2019-03-17.
- [19] Que es y para que sirve un barómetro ? - Ingeniería Mecafenix. <https://www.ingmecafenix.com/otros/medicion/barometro/>, Visitado: 2019-03-18.
- [20] What is Magnetometer: Types & Applications. <https://www.engineersgarage.com/articles/magnetometer>, Visitado: 2019-03-19.
- [21] How does GPS work?| Explore | physics.org. <http://www.physics.org/article-questions.asp?id=55>, Visitado: 2019-03-21.
- [22] Power Distribution Boards - How to choose the right one - Guides - DroneTrest. <https://www.dronetrest.com/t/power-distribution-boards-how-to-choose-the-right-one/1259>, Visitado: 2019-03-27.
- [23] Lo que hay que saber para elegir un equipo transmisor | Tienda y Tutoriales Arduino. <https://www.prometec.net/elegir-transmisor-receptor-radio/>, Visitado: 2019-03-29.
- [24] Telemetry Radios · PX4 User Guide. <https://docs.px4.io/en/telemetry/>, Visitado: 2019-04-01.
- [25] Motor EMAX MT2216-810KV CW + pareja helices 10x4.5 - pack motor & helices para drones. <https://rc-innovations.es/motor-Emax-MT2216-810KV-economicos-para-drones-con-pareja-helices>, Visitado: 2019-03-26.
- [26] Emaxmodel.com-R/C Store - GT2210. <https://www.emaxmodel.com/gt2210.html>, Visitado: 2019-03-26.
- [27] ZTW 4108 390KV Black Widow Con ESC Integrado - Multicopteros. <https://rc-innovations.es/Motores-multirrotores-aeromodelismo-brushless-drones/ZTW-motores-drones-multicopteros-esc-integrado/ztw-4108-390kv-black-widow-con-esc-integrado-ztw-4108-390kv-ztw>, Visitado: 2019-03-26.
- [28] Emaxmodel.com-R/C Store - EMX-MT-0070 MT3110 - 480KV 700KV CW Thread. <https://www.emaxmodel.com/mt3110.html>, Visitado: 2019-03-26.
- [29] APC 10X3.8SF- Helice APC 10X3.8SF giro normal. Incluye. <https://rc-innovations.es/Helices-APC-aviones-multirrotores-drones-alta-calidad/helice-APC-10X3.8SF-multicoptero-avion>, Visitado: 2019-03-25.
- [30] RCI Spider 30A OPTO "SmallSimonK 2013.05 - Estos ESCs fabricados por ZTW cuentan con una respuesta al a. <https://rc-innovations.es/Reguladores-ESC-UBEC-variadores-velocidad-motores-drones-multicoptero-multirrotores/RCI-Spider-ZTW-30A-OPTO-Small-multicoptero-simonk>, Visitado: 2019-03-29.
- [31] Controlador de velocidad de 30amp felpa TURNIGY w / BEC. https://hobbyking.com/es_es/turnigy-plush-30amp-speed-controller.html, Visitado: 2019-03-29.

- [32] ZTW Flash 30A BLHeli_S Oneshot125 - Oneshot42 - Multishot - Variadores para drones - Drones de carreras - FPV - Multicopteros - EFM8BB21F16. https://rc-innovations.es/Reguladores-ESC-UBEC-variadores-velocidad-motores-drones-multicoptero-multirrotores/ESC-ZTW-Flash-30a-blheli_s-oneshot125-oneshot42-multishot-variador, Visitado: 2019-03-29.
- [33] ZTW Spider 30A PRO Premium - BLheli oneshot125 - Variadores para drones - Drones de carreras - FPV - Multicopteros - Silab F390. <https://rc-innovations.es/Reguladores-ESC-UBEC-variadores-velocidad-motores-drones-multicoptero-multirrotores/ZTW-Spider-30a-PRO-Premium-blheli-oneshot125-Variador-drones>, Visitado: 2019-03-29.
- [34] DJI- Líder mundial en Drones/Cuadricópteros con cámara para Fotografía Aérea. <https://www.dji.com/es/naza-m-v2/spec>, Visitado: 2019-04-01.
- [35] ArduPilot Mega - Home. <https://www.ardupilot.co.uk/>, Visitado: 2019-04-02.
- [36] Pixhawk 4 · PX4 v1.8.2 User Guide. https://docs.px4.io/en/flight_controller/pixhawk4.html, Visitado: 2019-04-02.
- [37] Raspberry Pi 3 Model B - Raspberry Pi. <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>, Visitado: 2019-04-08.
- [38] 2812LED Controller & 2812ARM Light -4/-6 - Matek Systems. <http://www.mateksys.com/?portfolio=2812-ctrl#tab-id-2>, Visitado: 2019-05-13.
- [39] Camera Module V2 - Raspberry Pi. <https://www.raspberrypi.org/products/camera-module-v2/#buy-now-modal>, Visitado: 2019-04-10.
- [40] Connect ESCs and Motors — Copter documentation. <http://ardupilot.org/copter/docs/connect-escs-and-motors.html>, Visitado: 2019-05-14.
- [41] GPIO - Raspberry Pi Documentation. <https://www.raspberrypi.org/documentation/usage/gpio/>, Visitado: 2019-05-14.
- [42] Communicating with Raspberry Pi via MAVLink — Dev documentation. <http://ardupilot.org/dev/docs/raspberry-pi-via-mavlink.html>, Visitado: 2019-05-16.
- [43] ArduPilot :: About. <http://ardupilot.org/about>, Visitado: 2019-04-24.
- [44] Introduction · MAVLink Developer Guide. <https://mavlink.io/en/>, Visitado: 2019-04-24.
- [45] QGC - QGroundControl - Drone Control. <http://qgroundcontrol.com/>, Visitado: 2019-04-24.
- [46] Dronekit · PyPI. <https://pypi.org/project/dronekit/>, Visitado: 2019-04-24.
- [47] SITL Simulator (Software in the Loop) — Dev documentation. <http://ardupilot.org/dev/docs/sitl-simulator-software-in-the-loop.html>, Visitado: 2019-05-30.
- [48] Setting up SITL on Linux — Dev documentation. <http://ardupilot.org/dev/docs/setting-up-sitl-on-linux.html#setting-up-sitl-on-linux>, Visitado: 2019-05-30.
- [49] Messages (common) · MAVLink Developer Guide. <https://mavlink.io/en/messages/common.html>, Visitado: 2019-05-08.
- [50] SITL Advanced Testing — Dev documentation. <http://ardupilot.org/dev/docs/using-sitl-for-ardupilot-testing.html#using-sitl-for-ardupilot-testing>, Visitado: 2019-05-08.

- [51] Overview | NeoPixels on Raspberry Pi | Adafruit Learning System. <https://learn.adafruit.com/neopixels-on-raspberry-pi?view=all>, Visitado:2019-05-16.
- [52] 4. Advanced Recipes — Picamera 1.13 Documentation. <https://picamera.readthedocs.io/en/latest/recipes2.html#web-streaming>, Visitado:2019-05-16.